PRAGMATIC CONTEXTUALIZATION OF LANGUAGE IN LATENT COMMONSENSE

by

Rajkumar Pujari

A Dissertation

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Doctor of Philosophy

Department of Computer Science

West Lafayette, Indiana

August 2025

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Dan Goldwasser, Chair

Department of Computer Science

Dr. Jennifer Neville

Department of Computer Science

Dr. Clifton W. Bingham

Department of Computer Science

Dr. Ming Yin

Department of Computer Science

Dr. Ninghui Li

Department of Computer Science

Approved by:

Dr. Kihong Park

Head of the School Graduate Program

To my family, wife & all who inspire(d) me.

ACKNOWLEDGMENTS

I am truly grateful for all the guidance and support I received throughout my PhD program. I will elaborate on this when I submit my final thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Natural language has evolved over a long period as an efficient form of communication among humans. Linguistic discourse is usually created with a view of the consumer. Hence, it often omits a lot of information that is seemingly obvious to the consumer.

When designing machines that process and interact with natural language, it is necessary to account for contextualizing information. While sometimes it is simple to obtain and model such information, often this information is an aggregation of several tidbits of context drawn from a wide range of sources. These tidbits fit together nicely to complete the full picture.

In this thesis, we focus on designing, operationalizing, and evaluating frameworks that are capable of dynamically processing specific contextual information from a large corpus of noisy data. We work on two domains: US Political discourse and Chinese conversations.

We define the social and cultural context required in each domain. We formalize the structured organization of the contextual information. We propose frameworks and training paradigms that enable capturing contextual information effectively. We discuss the limitations of existing models on contextual information modeling.

We explore three main ideas in this thesis: (1) organizing and obtaining contextual information, (2) designing frameworks to encode text by joint conditioning on contextual information, and (3) creating evaluation tasks that measure the holistic contextual understanding capabilities of the models. We deal with the challenges of obtaining contextual information separately for each domain through extensive data collection and generation. We employ human expertise at each stage to enhance our data collection.

We evaluate the proposed framework and state-of-the-art NLP models quantitatively and qualitatively across several tasks. We present our findings which demonstrate the effectiveness of explicit context modeling compared to existing NLP models. We also benchmark human performance on these tasks which is much superior to the evaluated models.

Finally, we discuss a proposal to integrate the presented framework into the existing generative text model paradigm and extract the best of both worlds. We present some initial results which show promising improvements through this integration.

1. INTRODUCTION

Automated Natural Language Processing (NLP) has been a tantalizing challenge for decades. The crux of the challenge posed by NLP is resolving ambiguity in text. Ambiguous language is an organic feature of human communication that facilitates efficient exchange of information [1]. It manifests in several ways [2]: words with multiple meanings (*bank* of a river vs. financial *bank*), the way text is structured (*Old men and women were taken to safety.* - who is old? only men or both men & women?), physical settings (*The red car is mine.* - multiple red cars in the parking lot) and so on.

Contextual information in the form of *factual knowledge*, *commonsense*, *physical cues*, and *cultural nuances* often complements human comprehension of a text's meaning. For instance, in the US, the phrase '*How are you doing?*' is often used as a rhetorical greeting, while in other countries, it is typically a sincere inquiry about one's well-being.

Context could also come from a shared understanding of the situation. A case in point is political discourse on social media. In light of the recent BLM movement in the US, consider a tweet with the hashtag *#BlackLivesMatter* and a reply with *#BlueLivesMatter*. This exchange reflects how the individuals understand the circumstances and the political debate surrounding George Floyd's death caused by a police officer.

Several past efforts attempted to compile exhaustive repositories of commonsense knowledge for NLP [3–5]. But, while some commonsense knowledge could be explicitly stated (E.g: *the sun rises in the east*), a significant chunk of it is dynamically drawn or inferred from various sources in response to a presented situation. Humans recollect selected information and apply relevant components to fully comprehend the situation. For example, consider the context of a bill addressing the issue of mass shootings in US schools. When a *pro-gun control* Democratic politician calls for *protecting teachers*, we understand this as a plea to *regulate gun access to reduce mass shootings*. In contrast, consider the same plea from a Republican politician who is (1) publicly *pro-gun rights*, and (2) previously responded to a gun violence incident with '*only a good guy with a gun can stop bad guys with guns*'. Having come across the suggestion of '*arming teachers to protect against shooters*' from other right-

wing politicians, we could plausibly assume that the Republican politician is also signaling a similar approach.

Designing NLP systems that perform effectively on such tasks is hard. Traditionally such information is injected in various forms: rules [6], symbolic structure of the task [5, 7], statistical cues [8], annotated examples [9], feature design [10], and distributed embeddings [11]. Recently, since the advent of auto-regressive Large Language Models (LLMs) [12], such information could also be modeled through massive training on vast amounts of data or via instructions within prompts.

LLMs have demonstrated empirical effectiveness across a wide range of NLP tasks [13–15]. They have also shown interesting emergent properties that signal pragmatic understanding of language [16]. But, they still suffer from inherent limitations that hamper their ability to dynamically contextualize text like humans. LLM training data is static and re-training is often computationally expensive. Hence, new situation-specific knowledge like developing news can only be provided in the prompts. This limits the amount of context to the prompt size, which is often quite limited. They also tend to default to the statistically prominent patterns from the training corpus and are unable to adapt to niche cultural settings easily. This motivates the need for models that can explicitly process and effectively utilize contextual information at inference time.

Current NLP systems largely lack the ability to dynamically consume contextual information in bulk while performing pragmatic understanding tasks. In this thesis, we focus on developing frameworks (Ch. 2 & 4), evaluation tasks (Ch. 3), training paradigms (Ch. 2), and contextual information generation methods (Ch. 4) to address this gap.

The research questions we address in this thesis are:

RQ1: How do we obtain and organize a wide variety of contextual information that is required to understand nuanced ambiguity in text? (Ch. 2 & 4)

RQ2: How can this massive amount of information be used to create principled representations of text and various contextual components like entities, stances, preferences, and social norms? (Ch. 2)

RQ3: How do we operationalize tasks that require holistic social context grounding? How well do existing NLP models perform these tasks? What could enable them to do better? (Ch. 3)

RQ4: How do the proposed frameworks compare with state-of-the-art generative LMs? Can we improve LLMs by integrating proposed techniques into their architecture without losing the existing knowledge obtained from extensive pre-training? (Ch. 5)

We mainly focus on two domains in this thesis: US Political Discourse (Ch. 2 & 3) and Grounding Conversations in Culture-specific Social Norms (Ch. 4). We introduce important high-level ideas explored in this thesis briefly below:

Obtaining and Organizing Contextual Information: Contextual information might be dispersed over a huge corpus of data or is only latently observable in data without being explicitly stated. In the case of political discourse, news articles, previous discourse from politicians (tweets and press releases), their actions and associations (roll call voting patterns, campaign donors, etc), third-party analysis of politician behavior (issue-related grades of politicians), etc, provide crucial information.

In the context of cultural norm grounding, normative behaviors manifesting in several conversations paint a picture of tacit cultural agreements that are generally followed. Enumerating a complete set of descriptions of such behavior at scale using human-only annotations is impractical. Previous works propose an LLM-based generation of cultural norms [17–19]. However, LLMs are prone to hallucinations and biased against rarely occurring scenarios in the training data [20, 21]. Hence, we start with a large corpus of conversations and query LLMs to generate descriptions of norms that govern the observed behavior in the conversation. Building upon these generations, we employ a human-in-the-loop annotation protocol to filter invalid generations and identify clusters of descriptions that aggregate repeating behavior. We call them *norm concepts*.

Graph structures naturally lend to representing rich and connected information. Modeling contextual information requires careful design of the graph structures for each domain. We proposed graph schema structures for political discourse and cultural context in Ch. 2 and Ch. 4 respectively.

Designing Explicit Context Modeling Frameworks: Once we organize the contextual information into graphical schemas, we propose a novel neural architecture that unifies all the information in the graph in one shot. Our architecture generates a distributed representation for each item in the graph that is contextualized by the representations of others. It can dynamically respond to queries, focusing the induced representation on a specific context. Further, in Ch. 5, we propose an integration of this architecture into transformer-based auto-regressive LLM.

Designing Training Paradigms: Using the contextual graphs and novel architecture for explicit context modeling, we propose carefully designed link-prediction style learning tasks. We show that these learning tasks work best in conjunction with the proposed architecture when compared to a competent baseline architecture trained on the same data.

Operationalizing Social Context Grounding Evaluation: We propose a suite of evaluation tasks that require a joint understanding of multiple sources of context to be effectively solved. We collect datasets for these tasks through a hybrid approach of human-expert annotation and machine-in-the-loop data augmentation. We show that these tasks are relatively straightforward for humans with knowledge of the events. We further observe that existing NLP models lag significantly behind human performance on these datasets. We also show that explicit context models outperform much larger models on these tasks while still lagging behind human performance.

2. DISCOURSE CONTEXTUALIZATION FRAMEWORK

Over the last decade, political discourse has moved from traditional outlets to social media. This process, starting in the '08 U.S. presidential elections, has peaked in recent years, with former president Trump announcing the firing of top officials as well as policy decisions over Twitter. This presents a new challenge to the NLP community, *how can this massive amount of political content be used to create principled representations of politicians, their stances on issues and legislative preferences?*

This is not an easy challenge as in political texts perspective is often subtle rather than explicit [22]. Choices of mentioning or omitting certain entities or attributes can reveal the author's agenda. For example, tweeting "*mass shootings are due to a huge mental health problem*" in reaction to a mass shooting is likely to be indicative of opposing gun control measures, despite the lack of an explicit stance in the text.

Recent advances in Pretrained Language Models (PLMs) in NLP [23–25] have greatly improved word representations via contextualized embeddings and powerful transformer units, however such representations alone are not enough to capture nuanced biases in political discourse. Two of the key reasons are: (i) they do not directly focus on entity/issue-centric data and (ii) they only represent linguistic context rather *external* political context.

Our main insight is that effectively detecting such bias from text requires modeling the broader political context of the document. This can include understanding relevant facts related to the event addressed in the text, the ideological leanings and perspectives expressed by the author in the past, and the sentiment/attitude of the author towards the entities referenced in the text. We suggest that this holistic view can be obtained by combining information from multiple sources, which can be of varying types, such as news articles, social media posts, quotes from press releases and historical beliefs expressed by politicians.

For example, consider the following tweet in context of a school shooting: *We need to treat our teachers better! We should keep them safe.* If the author of the tweet is Kamala Harris (known to be pro-gun control), this tweet is likely to be understood as "*ban guns to avoid mass shootings in schools*". However, if the same tweet is from Mike Pence, whose stance on *guns* is: "*firearms in the hands of law abiding citizens makes our communities*

safer", the tweet could mean "*arming school teachers stops active shooters*". This example demonstrates that depending on the context, the same text could signal completely different real-world actions. Hence, we need to model the broader context of the text in order to understand its true meaning. Visualization projecting the tweet representation into a 2D space is given in figure 2.1, and shows how contextualization from our model helps disambiguate this example. First, we show the BERT-base representation of the tweet (`Tweet-BERT`). We also show the BERT-base representations of the known stances of Pence and Harris on gun control (`{Mike Pence,Kamala Harris} Stance-BERT`). Finally, we apply our model, contextualizing the ambiguous tweet representation with speaker information (`{Mike Pence,Kamala Harris} Tweet-Contextualized`). The visualization captures how this representation can disambiguate the different interpretations of the same text, and capture their differences.

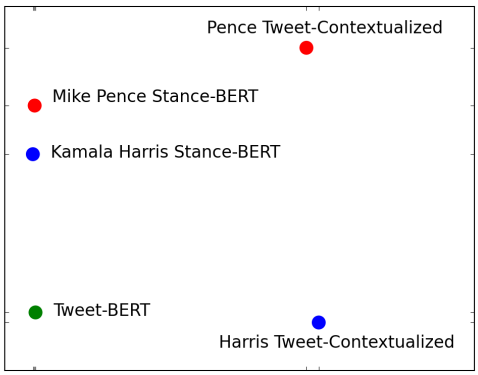


Figure 2.1. BERT vs. Author-Contextualized Encoder Composer Representation of an Ambiguous Tweet

A computational setting for this approach, *combining text and context analysis*, requires two necessary attributes: (i) an *input* representation that combines all the different types of information meaningfully and (ii) the ability to create *a meaningful unified representation* in one-shot, that captures the complementary strengths of the different inputs.

We address the first challenge by introducing a graph structure that ties together first-person informal (tweets) and formal discourse (press releases and perspectives), third-person current (news) and consolidated (Wikipedia) discourse. These documents are connected

via their authors, the issues/events they discuss and the entities mentioned in them. As a clarifying example consider the tweet by former-President Trump *"The NRA is under siege by Cuomo"*. This tweet will be represented in our graph by connecting the text node to the author node (Trump) and the referenced entity node (NY Gov. Cuomo). These settings are shown in Fig. 2.2.

We propose a novel neural architecture that unifies all the information in the graph in one-shot. *Our architecture generates a distributed representation for each item in the graph that is contextualized by the representations of others.* It can dynamically respond to queries, focusing the induced representation on a specific context. In our example, this results in a modified tweet representation helping us characterize Trump's opinion of Cuomo *in the context of the guns issue*. Our architecture consists of an *Encoder* combining all documents related to a given node to generate an initial node representation and a *Composer*, a Graph Attention Network (GAT), composing the graph structure to generate contextualized node embeddings.

We design two self-supervised learning tasks to train the model and capture structural dependencies over the rich discourse representation, predicting *Authorship* and *Referenced Entity* links over the graph structure. Intuitively, the model is required to understand subtle language usage; *Authorship* prediction requires the model to differentiate between: (i) the language of one author from another and (ii) the language of the author in context of one issue vs another issue. *Referenced Entity* prediction requires understanding the language used by a specific author when discussing a particular entity, given the author's past discourse.

We focus on a specific graph element–*politicians*, and evaluate their resulting discourse representation on several empirical tasks which capture their stances and preferences. Our evaluation demonstrates the importance of each component of our model and usefulness of the learning tasks. To summarise, our research contributions include:

1. A novel graphical structure connecting various types of documents, entities, issues and events.
2. An effective neural architecture, *Compositional Reader*, processing all information in one-shot, and designing two effective tasks for training it.

3. Designing & performing quantitative and qualitative evaluation showing that our graph structure, neural architecture and learned representations are meaningful and effective for representing politicians and their stances on issues.[1]

2.1 Related Work

Due to recent advances in text representations catalysed by [26], [27] and followed by [23], [25] and [24], we are now able to create rich textual representations, effective for many NLP tasks. Although contextual information is captured by these models, they are not explicitly designed to capture entity/event-centric information. Hence, tasks that require such information [28–32], would benefit from more focused representations.

Of late, several works attempted to solve such tasks, such as analyzing relationships and their evolution [33, 34], analyzing political discourse on news and social media [35, 36] and political ideology [37–39]. Various political tasks such as roll call vote prediction [40–44], entity stance detection [45, 46], hyper-partisan/fake news detection [47–49] require a rich understanding of the context around the entities that are present in the text. But, the representations used are usually limited in scope to specific tasks and not rich enough to capture information that is useful across several tasks.

The Compositional Reader model, that builds upon [23] embeddings and consists of a transformer-based Graph Attention Network inspired from [50] and [51], aims to address those limitations via a generic entity-issue-event-document graph, which is used to learn highly effective representations.

Representing legislative preferences is typically done by modeling the ideal point of legislators represented in a Euclidean space from roll-call records [52]. Recent approaches incorporate bill text information into this representation [53–56]. Most relevant to our work is [57] which uses social media information. We significantly extend these approaches by contextualizing the social media content using a novel architecture.

Data	Count	Data	Count
News Events	367	Tweets	86,409
Author Entities	455	Press Releases	62,257
Ref. Entities	10,506	Perspectives	30,446
Wikipedia	455	News Articles	8,244
Total Docs	187,811		

Table 2.1. Summary statistics of data

2.2 Data

We collected US political text related to 8 broad topics: *guns, LGBTQ rights, abortion, immigration, economic policy, taxes, middle east & environment*. The data focused on 455 members of the US Congress. We collected political text data relevant to above topics from 5 sources: press statements by political entities from ProPublica Congress API[2], Wikipedia articles describing political entities, tweets by political entities (Congress Tweets, [58]), perspectives of the senators and congressmen regarding various political issues from ontheissues.org and news articles & background of the those political issues from allsides.com. A total of 187,811 documents were used to train our model, as shown in Tab. 2.1.

2.2.1 Event Identification

To identify news events, we use news article headlines. We find the mean (μ) and standard deviation (σ) of the number of articles published per day for each issue. If more than $\mu + \sigma$ number of articles are published on a single day for a given issue, we identify it as the beginning of an event. Then, we skip 7 days and look for a new event.

In our setting, events *within* an issue are non-overlapping. We divide events for each issue separately, hence events for different issues do overlap. These events last for $7 - 10$ days on average and hence the non-overlapping assumption within an issue is a reasonable relaxation of reality. To illustrate our point: coronavirus and civil-rights are separate issues and hence have overlapping events. An example event related to coronavirus could be "First case of

[1]↑Repository: https://github.com/pujari-rajkumar/compositional_learner
[2]↑https://projects.propublica.org/api-docs/congress-api/

COVID-19 outside of China". Similarly an event about civil-rights could be "Officer part of George Floyd killing suspended". We inspected the events manually by random sampling. More example events are in the appendix.

2.2.2 Data Pre-processing

We use Stanford CoreNLP tool [59], Wikifier [60] and BERT-base-uncased implementation by [61] to preprocess data for our experiments. We tokenize the documents, apply coreference resolution and extract referenced entities from each document. The referenced entities are then wikified using Wikifier tool [60]. The documents are then categorized by issues and events. News articles from allsides.com and perspectives from ontheissues.org are already classified by issues. We use keyword based querying to extract issue-wise press releases from Propublica API. We use hashtag based classification for tweets. A set of gold hashtags for each issue was created and the tweets were classified accordingly[3]. Sentence-wise BERT-base embeddings of all documents are computed.

2.2.3 Query Mechanism

We implemented a query mechanism to obtain relevant subsets of data from the corpus. Each query is a triplet of *entities, issues & lists of event indices corresponding to each of the issues*. Given a query triplet, news articles related to the events for each of the issues, Wikipedia articles for each of the entities, background descriptions of the issues, perspectives of each entity regarding each of the issues and tweets & press releases by each of the entities related to the events in the query are retrieved. Referenced entities for each of the sentences in documents and sentence-wise BERT embeddings of the documents are also retrieved.

2.3 Compositional Reader

In this section, we describe the architecture of the proposed 'Compositional Reader' model in detail. It contains 3 key components: Graph Generator, Encoder and Composer. Given a query output of the query mechanism from Sec. 2.2.3, Graph Generator creates

[3]↑Data collection is detailed in appendix

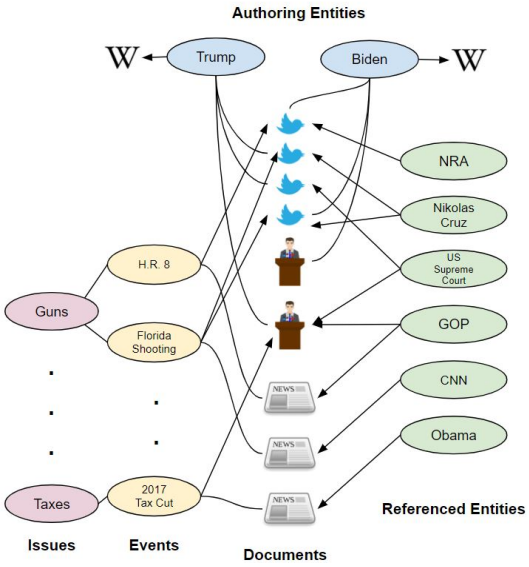


Figure 2.2. Example Text Graph from Graph Generator

a directed graph with entities, issues, events and documents as nodes. Encoder is used to generate initial node embeddings for each of the nodes. Composer is a transformer-based Graph Attention Network (GAT) followed by a pooling layer. It generates the final node embeddings and a single summary embedding for the query graph. Each component is described below.

2.3.1 Graph Generator

Given the output of the query mechanism for a query, the Graph Generator creates a directed graph with 5 types of nodes: authoring entities, referenced entities, issues, events and documents. Directed edges are used by Composer to update source node representations using destination nodes. We design the topology with the main goal of capturing the representations of events, issues and referenced entities that reflect author's opinion about them. We add edges from issues/events to author's documents but omit the other direction as our main goal is to contextualize issues/events using author's opinions.

Bidirectional edges from authors to their Wikipedia articles, tweets, press releases and perspectives, from issues to their background description, events and from events to news

articles describing them are added. Uni-directional edges from events to tweets and press releases, from issues to author perspectives and from referenced entities to the documents that mention them are added. An example graph is shown in Fig. 2.2.

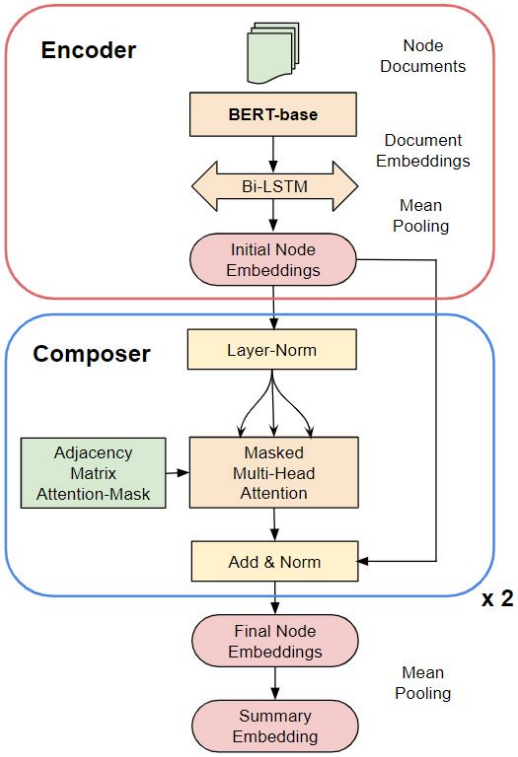


Figure 2.3. Encoder-Composer Architecture

2.3.2 Encoder

Encoder is used to compute the initial node embeddings. It consists of BERT followed by a Bi-LSTM. For each node, it takes a sequence of documents as input. The documents are ordered temporally. The output of Encoder is a single embedding of dimension d_m for each node. Given a node $\mathcal{N} = \{D_1, D_2, \ldots, D_d\}$ consisting of d documents, for each document D_i, contextualized embeddings of all the tokens are computed using BERT. Token embeddings are computed sentence-wise to avoid truncating long documents. Then, token embeddings of each document are mean-pooled to get the document embeddings $\vec{\mathcal{N}}^{bert} = \{\vec{D_1}^{bert}, \vec{D_2}^{bert}, \ldots, \vec{D_d}^{bert}\}$ where $\vec{D_\text{i}}^{bert} \in \mathbb{R}^{1 \times d_m}$, d_m is the dimension of a BERT token embedding. The

sequence $\vec{\mathcal{N}}^{bert}$ is passed through a Bi-LSTM to obtain an output sequence $\vec{E} = \{\vec{e_1}, \vec{e_2}, \ldots, \vec{e_d}\}$, $\vec{e_i} \in \mathbb{R}^{1\times h}$, where $h/2$ is the hidden dimension of the Bi-LSTM, we set $h = d_m$ in our model. Finally, the output of Encoder is computed by mean-pooling the sequence $\vec{E}$. We use BERT-base-uncased model in our experiments where $d_m = h = 768$. Initial node embeddings of all the document nodes are set to Encoder output of the documents themselves. For authoring entity nodes, their Wikipedia descriptions, tweets, press releases and perspective documents are passed through Encoder. For issue nodes, background description of the issue is used. For event nodes, all the news articles related to the event are used. For referenced entities, all documents that mention the entity are used.

2.3.3 Composer

Composer is a transformer-based graph attention network (GAT) followed by a pooling layer. We use the transformer encoding layer proposed by [27], without the position-wise feed forward layer, as graph attention layer. Position-wise feed forward layer is removed as in contrast with sequence-to-sequence prediction tasks, nodes in a graph usually have no ordering relationship between them. Adjacency matrix of the graph is used as the attention mask. Self-loops are added for all nodes so that updated representation of the node also depends on its previous representation. Composer module uses $l = 2$ graph attention layers in our experiments. Composer module generates updated node embeddings $\mathbb{U} \in \mathbb{R}^{n\times d_m}$ and a summary embedding $\mathbb{S} \in \mathbb{R}^{1\times d_m}$ as outputs. The output dimension of node embeddings is 768. Equations that describe Composer unit are:

$$
\begin{aligned}
&\mathbb{E} \in \mathbb{R}^{d_m \times n}, \mathcal{A} \in \{0,1\}^{n\times n} \\
&\mathbb{G} = LN(\mathbb{E}) \\
&Q = W_q^T\mathbb{G}, K = W_k^T\mathbb{G}, V = W_v^T\mathbb{G} \\
&M = \frac{Q^T K}{\sqrt{d_k}}, M = mask(M, \mathcal{A}) \\
&\mathbb{O} = MV^T, \mathbb{U} = W_o^T\mathbb{O} + \mathbb{E} \\
&\mathbb{S} = \textit{mean-pool}(\mathbb{U})
\end{aligned}
\tag{2.1}
$$

where n is number of nodes in the graph, d_m is the dimension of a BERT token embedding, d_k, d_v are projection dimensions, n_h is number of attention heads used and $Q \in \mathbb{R}^{n_h \times d_k \times n}$, $K \in \mathbb{R}^{n_h \times d_k \times n}$, $V \in \mathbb{R}^{n_h \times d_v \times n}$, $\mathbb{O} \in \mathbb{R}^{n_h d_v \times n}$, $M \in \mathbb{R}^{n_h \times n \times n}$. $W_q \in \mathbb{R}^{d_m \times n_h d_k}$, $W_k \in \mathbb{R}^{d_m \times n_h d_k}$, $W_v \in \mathbb{R}^{d_m \times n_h d_v}$ and $W_o \in \mathbb{R}^{n_h d_v \times d_m}$ are weight parameters to be learnt. $\mathbb{E} \in \mathbb{R}^{d_m \times n}$ is the output of the encoder. $\mathcal{A} \in \{0,1\}^{n \times n}$ is the adjacency matrix. We set $n_h = 12$ and $d_k = d_v = 64$.

2.4 Learning Tasks

We design two learning tasks to train the Compositional Reader model: *Authorship Prediction* and *Referenced Entity Prediction*. Both the tasks are intuitively designed to train the model to learn the association between the author node representation and the language used by the particular author. These tasks are two variations of link prediction over the graph. The tasks are detailed below.

2.4.1 Authorship Prediction

Authorship Prediction is designed as a binary classification task. In this task, the model is given a graph generated by the graph generator in subsection 2.3.1, an author node and a document node. The task is to predict whether or not the document was authored by the input author.

Intuition behind this learning task is to enable our model to learn differentiating between: 1) language of an author's first-person discourse vs. third person discourse in news articles, 2) language of an author vs. language used by other authors and 3) language of an author in context of one issue vs. in context of other issues. The model sees documents by the author in the graph and learns to decide whether or not the input document is by the same author and talking about the same issue.

Data Training data for the task was created as follows: for a particular author-issue pair, we obtain a data graph similar to Fig. 2.2 using the query mechanism in subsection 2.2.3. To create a positive data sample, we sample a document d_{i} authored by the entity a_{i} and remove the edges between the nodes a_{i} and d_{i}. Negative samples were designed carefully

Model	IS Acc	IS F1	OS Acc	OS F1
Authorship Prediction				
BERT Adap.	93.01	92.31	95.56	95.20
Comp. Reader	99.49	99.47	99.42	99.39
Reference Entity Prediction				
BERT Adap.	76.57	75.21	76.26	73.67
Comp. Reader	78.52	77.51	78.98	78.62

Table 2.2. Learning Tasks In-Sample & Out-Sample Results on Test Data. Acc.denotes Accuracy. F1 Score for the Positive Class is Reported.

in 3 batches to align with our above task objectives. In the first batch, we sample news article nodes from the same graph. In the second batch, we obtain tweets, press releases and perspectives of the same author but from a different issue. In the third batch, we sample documents related to the same issue but from other authors. We generate $421,284$ samples in total, with $252,575$ positive samples and $168,709$ negative samples. We randomly split the data into training set of $272,159$ samples, validation set of $73,410$ samples and test set of $75,715$ samples.

Architecture We concatenate the initial and final node embeddings of the author, document and also the summary embedding of the graph to obtain inputs to the fine-tuning layers for Authorship Prediction task. We add one hidden layer of dimension 384 before the classification layer.

Out-sample Evaluation We perform out-sample experiments to evaluate generalization capability to unseen author data. We train the model on training data from two-thirds of politicians and test on the test sets of others. Results are shown in Tab. 2.2.

Graph Trimming We perform graph trimming to make the computation tractable on a single GPU. We randomly drop 80% of the news articles, tweets and press releases that are not related to the event to which d_{i} belongs. We use graphs with 200-500 nodes and batch size of 1.

2.4.2 Referenced Entity Prediction

This is also a binary classification task. Given a data graph, a document node with a masked entity and a referenced entity node the graph, the task is to predict whether the referenced entity is same as the masked entity. Intuition behind this learning task is to enable our model to learn the correlation between the language of the author in the document and the masked entity. For example, in context of recent Donald Trump's impeachment hearing, consider the sentence 'X needs to face the consequences of their actions'. Depending upon the author, X could either be '*Donald Trump*' or '*Democrats*'. Learning to understand such correlations by looking at other documents from the same author is effective in capturing meaningful author representations.

Data To create training data, we sample a document from the data graph. We mask the most frequent entity in the document with a generic <ENT> token. We remove the link between the masked entity and the document in the data graph. We sample another referenced entity from the graph to generate a negative example. We generated $252,578$ samples for this task, half of them positive. They were split into $180,578$ training samples, validation and test sets of $36,400$ samples each.

Architecture We use fine-tuning architecture similar to Authorship Prediction on top of Compositional Reader for this task as well. We keep separate fine-tuning parameters for each task as they are fundamentally different prediction problems. Compositional Reader is shared. We apply graph trimming for this task as well. We also perform out-sample evaluation for this learning task.

Results Performance of the BERT Adaptation baseline and Compositional Reader model are shown in Tab 2.2. On Authorship Prediction, out-sample performance doesn't drop for either model. This shows the usefulness of our graph formulation which allows the models to learn linguistic nuances. On Referenced Entity Prediction, F1 score for our model improves from 77.51 from in-sample to 78.62 on out-sample while BERT adaptation baseline's F1 drops slightly from 75.21 to 73.67

Model	Paraphrase All Grades	Paraphrase A/F Grades	NRA Val Acc	NRA Test Acc	LCV Val Acc	LCV Test Acc
BERT	41.55%	38.52%	55.93 ± 0.72	54.83 ± 1.79	54.28 ± 0.31	52.63 ± 1.21
BERT Adap.	37.54%	42.62%	71.23 ± 3.93	69.95 ± 3.33	60.58 ± 1.56	59.09 ± 1.77
Encoder	56.16%	48.36%	83.95 ± 1.24	81.34 ± 0.86	65.10 ± 0.46	63.42 ± 0.35
Comp. Reader	63.32%	63.93%	84.19 ± 0.98	81.62 ± 1.23	65.55 ± 1.33	62.24 ± 0.56

Table 2.3. Results of *Grade Paraphrase* and *Prediction* tasks. Acc denotes Accuracy, NRA and LCV denote Grade Prediction tasks. Mean ± Std. Dev for 5 random seeds for Grade Prediction showing statistical significance.

Session	Majority Class (%)	Accuracy (%)		Precision (%)		Recall (%)		F1 (%)	
		NW-GL	CR	NW-GL	CR	NW-GL	CR	NW-GL	CR
106	83.23	85.04	85.65	91.89	91.67	90.22	91.27	91.05	91.47
107	85.78	87.62	88.30	90.12	89.48	95.37	97.17	92.67	93.16
108	87.02	92.03	92.27	93.46	93.52	97.59	97.83	95.48	95.32
109	83.57	85.42	87.23	88.38	88.39	93.84	97.33	91.49	92.65
Average	84.90	87.53	88.36	90.96	90.77	94.26	95.90	92.67	93.15

Table 2.4. Roll Call Prediction Results. NW-GL represents the best performing model of [42] as replicated by us using their official implementation. CR represents Compositional Reader results. The improvements are statistically significant as per McNemar's test.

2.5 Evaluation

We evaluate our model and pre-training tasks in a systematic manner using several quantitative tasks and qualitative analysis. Quantitative evaluation includes *Grade Paraphrase* task, *Grade Prediction* on *National Rifle Association (NRA)* and *League of Conservation Voters (LCV)* grades data followed by *Roll Call Vote Prediction* task. Qualitative evaluation includes entity-stance visualization for issues and Opinion Descriptor Generation. We compare our model's performance to BERT representations, the BERT adaptation baseline and representations from the Encoder module. Baselines and the evaluation tasks are detailed below. Further evaluation tasks are in the appendix.

2.5.1 Baselines

BERT: We compute the results obtained by using pooled BERT representations of relevant documents for each of the quantitative tasks. Details of the chosen documents and

the pooling procedure is described in the relevant task subsections. We chose BERT-base over BERT-large due to the complexity of running the learning tasks on embedding dimension 768 vs 1024. A bigger embedding dimension results in lesser context (lesser number of nodes in the graph).

Encoder Representations: We compare the performance of our model to the results obtained by using initial node embeddings generated from the Encoder for each of the quantitative tasks.

BERT Adaptation Model: We design a BERT adaptation baseline for the learning tasks. BERT adaptation architecture is same as the Encoder of the Compositional Reader model. While Encoder's parameters are trained via back-propagation through the Composer, BERT adaptation model is directly trained on learning tasks. In BERT adaptation, once we generate the data graph, we pass the mean-pooled sentence-wise BERT embeddings of the node documents through a Bi-LSTM. We mean-pool the output of Bi-LSTM to get node embeddings. We use fine-tuning layers on top of thus obtained node embeddings for both the learning tasks. BERT Adaptation baseline allows us to showcase the importance of our proposed training tasks via comparison with BERT-base representations. It also demonstrates the usefulness of Composer.

2.5.2 Grade Paraphrase Task

National Rifle Association (NRA) assigns letter grades (A+, A, ..., F) to politicians based on candidate questionnaire and their gun-related voting. We evaluate our representations on their ability to predict these grades. We collected the historical data of politicians' NRA grades from everytown.org.

In *Grade Paraphrase* task, we evaluate our representations directly *without* training on the NRA data. Grades are divided into two classes: grades including and above $B+$ are in positive class and grades from $C+$ to F are clustered into negative. We formulate representative sentences for them:

- POSITIVE: *I strongly support the NRA*
- NEGATIVE: *I vehemently oppose the NRA*

For each politician, we obtain data graph for the issue *guns*. We input the data graph to Compositional Reader model and use the node embeddings of the author politician ($\vec{n}_{auth}$), issue *guns* ($\vec{n}_{guns}$) and referenced entity *NRA* ($\vec{n}_{NRA}$). For some politicians, $\vec{n}_{NRA}$ is not available as they have not referenced *NRA* in their discourse. We just use $\vec{n}_{auth}$ and $\vec{n}_{guns}$ for them. We compute BERT-base embeddings for the representative sentences to obtain $\vec{pos}_{NRA}$ and $\vec{neg}_{NRA}$. We mean-pool the three embeddings $\vec{n}_{auth}$, $\vec{n}_{guns}$ and $\vec{n}_{NRA}$ to obtain $\vec{n}_{stance}$. We compute cosine similarity of $\vec{n}_{stance}$ with $\vec{pos}_{NRA}$ & $\vec{neg}_{NRA}$. Politician is assigned the higher similarity class.

We compare our model's results to BERT-base, BERT adaptation and Encoder embeddings. For BERT-base, we compute $\vec{n}_{stance}$ by mean-pooling the sentence-wise BERT embeddings of tweets, press releases and perspectives of the author on all events related to the issue *guns*. Results are shown in Tab. 2.3. Compositional Reader achieves 63.32% accuracy. Encoder embeddings get 56.16%. Mean-pooled BERT-base embeddings get 41.55%. Using node embeddings from BERT adaptation model yields 37.54%. When we evaluate using only 'A'/'F' grades, we obtain 63.93% accuracy for Compositional Reader, 48.36% for Encoder, 42.62% for BERT adaptation and 38.52% for BERT-base.

2.5.3 Grade Prediction Task

NRA Grades This is designed as a 5-class classification task for grades {*A, B, C, D & F*}. We train a simple feed-forward network with one hidden layer. The network is given 2 inputs $\vec{n}_{auth}$ & $\vec{n}_{guns}$. When $\vec{n}_{NRA}$ is available for an author, we set $\vec{n}_{guns} = mean(\vec{n}_{NRA}, \vec{n}_{guns})$. The output is a binary prediction.

We perform $k = 10$-fold cross-validation for this task. We repeat the entire process for 5 random seeds and report the results with confidence intervals. We perform this evaluation for BERT-base, BERT adaptation, Encoder and Compositional Reader. To compute $\vec{n}_{auth}$ for BERT-base, we mean-pool the sentence-wise embeddings of all author documents on *guns*. For $\vec{n}_{guns}$, we use the background description document of issue *guns*. Results on the test set are in Tab. 2.3.

LCV Grades This is similar to NRA Grade Prediction task. This is a 4-way classification task. *League of Conservation Voters* (LCV) assigns a score ranging between 0-100 to each politician depending upon their environmental voting activity. We segregate politicians into 4 classes ($0-25$, $25-50$, $50-75$, $75-100$). We obtain input to the prediction model by concatenating $\vec{n}_{auth}$ and $\vec{n}_{environment}$. We use same fine-tuning architecture as NRA Grade Prediction task.

Results of Grade Prediction task are shown in Tab. 2.3. On *NRA Grade Prediction*, which is a 5-way classification task, our model achieves an accuracy of 81.62 ± 1.23 on the test set. Our model outperforms BERT representations by 26.79 ± 3.02 absolute points on the test set. On *LCV Grade Prediction* task which is a 4-way classification, our model achieves 9.61 ± 1.77 point improvement over BERT representations.

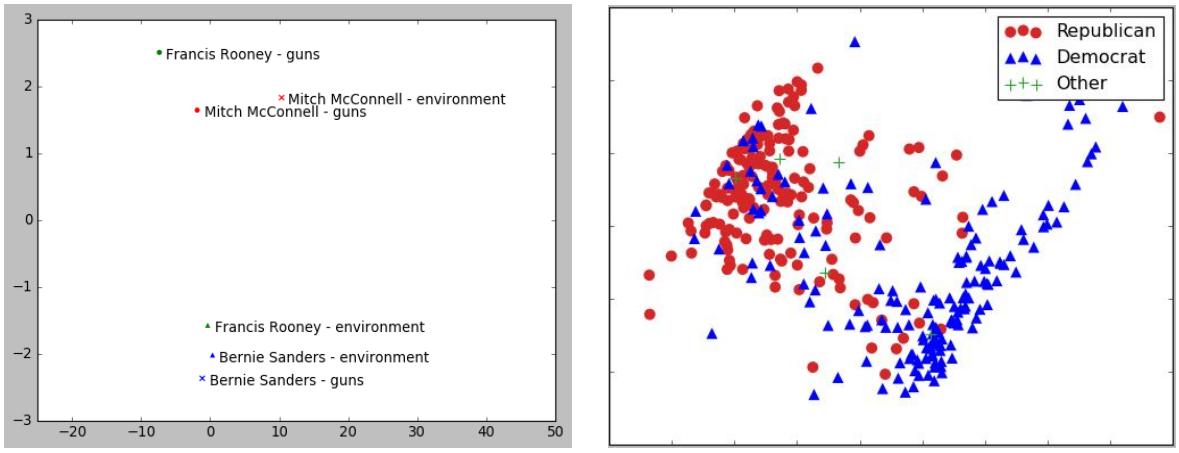
2.5.4 Roll Call Vote Prediction Task

This task was proposed in [42]. We skip the finer details of the task for brevity. The task aims to predict the voting behaviour of US politicians on roll call votes. Given the bill texts and voting history of the politicians, the aim is to predict future voting patterns of the politicians. We inject our politician author embeddings from Compositional Reader model to improve the performance on the task. We input all the politician first-person discourse from our data to compute politician author embeddings using Compositional Reader model. We use these embeddings to initialize the legislator embeddings in their news-augmented glove model, which is their best performing model. We use the data splits provided in their official implementation. We use their code to reproduce their results. Results are shown in Tab. 2.4.

2.5.5 Qualitative Evaluation

Politician Visualization We perform Principle Component Analysis (PCA) on issue embeddings ($\vec{n}_{issue}$) of politicians obtained using the same method as in NRA Grade prediction. We show one such interesting visualization in Fig. 2.4. Sen. McConnell, a Republican who expressed right-wing views on both *environment* and *guns*. Sen. Sanders, a Democrat that

expressed left-wing views on both. Rep. Rooney, a Republican who expressed right-wing views on *guns* but left-wing views on *environment*. Fig. 2.4 demonstrates that this information is captured by our representations. Additional such visualizations are included in the appendix.



(a) Individual Stances (b) Issue *Guns*

Figure 2.4. PCA Visualizations of Politician Embeddings

Issue	Opinion Descriptors	Issue	Opinion Descriptors
Mitch McConnell	Republican	Nancy Pelosi	Democrat
abortion	fundamental, hard, eligible, embryonic, unborn	*abortion*	future, recent, scientific, technological, low
environment	achievable, more, unobjectionable, favorable, federal	*environment*	forest, critical, endangered, large, clear
guns	substantive, meaningful, outdone, foreign, several	*guns*	constitutional, ironclad, deductible, unlawful, fair
immigration	federal, sanctuary, imminent, address, comprehensive	*immigration*	immigrant, skilled, modest, overall, enhanced
Donald Trump	Republican	Joe Biden	Democrat
guns	terrorist, public, ineffective, huge, inevitable, dangerous	*guns*	banning, prohibiting, ban, maintaining, sold
immigration	early, dumb, birthright, legal, difficult	*taxes*	progressive, economic, across-the-board, annual, top

Table 2.5. Opinion Descriptor Labels for Politicians. They show the most representative adjectives used by the politicians in context of each issue.

Model	Accuracy	Model	Accuracy
Comp.Reader	63.32%		
w/o Tweets	63.32%	Only Tweets	40.11%
w/o Press	63.04%	Only Press	55.87%
w/o Persp.	59.31%	Only Persp.	60.74%

Table 2.6. Ablation Study on *Grade Paraphrase* task for various types of documents

Issue Visualization We present visualization of politicians on the issue *guns* in Fig. 2.4. We observe that *guns* tends to be a polarizing issue. This shows that our representations are

able to effectively capture relative stances of politicians. We observe that issues that have traditionally had clear conservative vs liberal boundaries such as *guns* & *abortion* are more polarized compared to issues that evolve with time such as *middle-east* & *economic-policy*. Visualization for issue *abortion* is in the appendix.

2.5.6 Opinion Descriptor Generation

This task demonstrates a simple way to interpret our contextualized representations as natural language descriptors. It is an unsupervised qualitative evaluation task. We generate opinion descriptors for authoring entities for specific issues. We use the final node embedding of the issue node ($\vec{n}_{issue}$) for each politician to generate opinion descriptors.

Inspired from [34], we define our candidate space for descriptors as the set of adjectives used by the entity in their tweets, press releases and perspectives related to an issue. Although [34] uses verbs as relationship descriptor candidates, we opine that adjectives describe opinions better. We compute the representative embedding for each descriptor by mean-pooling the contextualized embeddings of that descriptor from all its occurrences in the politician's discourse. This is the one of the key differences with prior descriptor generation works such as [34] and [33]. They work in a static word embedding space. But, our embeddings are contextualized and also reside in a higher dimensional space. In an unsupervised setting, this makes it more challenging to translate from distributional space to natural language tokens. Hence, we restrict the candidate descriptor space more than [34] and [33]. We rank all the candidate descriptors according to cosine similarity of its representative embedding with the vector $\vec{n}_{issue}$.

We present some of the results in Tab. 2.5. In contrast to [33] and [34], our model doesn't need the presence of both the entities in text to generate opinion descriptors. This is often the case in first person discourse. Results are shown in table 2.5.

2.5.7 Ablation Study

Further, we investigate the importance of various components. We perform ablation study over various types of documents on the *NRA Grades Paraphrase* task. the results

are shown in Tab. 2.6. They indicate that *perspectives* are most useful while *tweets* are the least useful documents for the task. As *perspectives* are summarized ideological leanings of politicians, it is intuitive that they are more effective for this task. Tweets are informal discourse and tend to be very specific to a current event, hence they are not as useful for this task.

2.6 Conclusion

We tackle the problem of *understanding politics*, i.e., creating unified representations of political figures capturing their views and legislative preferences, directly from raw political discourse data originating from multiple sources. We propose the Compositional Reader model that composes multiple documents in one shot to form a unified political entity representation, while capturing the real-world context needed for representing the interactions between these documents.

We evaluate our model on several qualitative and quantitative tasks. We outperform BERT-base model on both types of tasks. Our qualitative evaluation demonstrate that our representations effectively capture nuanced political information.

3. LANGUAGE IN CONTEXT

Over the past decade, micro-blogging websites have become the primary medium for US politicians to interact with general citizens and influence their stances for gaining support. As a result, politicians from the same party often coordinate the phrasing of their social messaging, to amplify their impact [62, 63]. Hence, repetitive, succinct phrases, such as *"Thoughts and Prayers"*, are extensively used, although they signal more nuanced stances. Moreover, the interaction among politicians from opposing parties often leads to messaging phrased similarly, but signaling opposing real-world actions. For example, *'Thoughts and Prayers'*, when used by Republicans, expresses condolences in mass shooting events, but when used by Democrats conveys an *angry* or *sarcastic* tone as a call for action demanding *"tighter gun control measures"*. Similarly, fig. 3.1 shows contrasting interpretations of the phrase "*We need to keep our teachers safe*!" depending on different speakers and in the context of different events.

Humans familiar with the stances of a politician and, possessing knowledge about the event from the news, can easily understand the *intended meaning* of political phrases. However, automatically understanding such language is challenging. Our main question in this

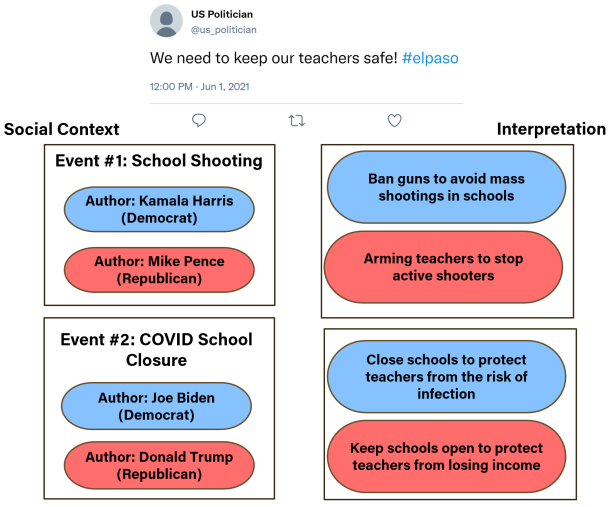


Figure 3.1. An example of varied *intended meanings* behind the same political message depending on the Author and Event in context

paper is - ***Can an NLP model find the right meaning?*** From a linguistic perspective, we follow the distinction [64] between *semantic* interpretation (i.e., meaning encoded directly in the utterance and does not change based on its external context), and *pragmatic* interpretation (that depends on extra-linguistic information). The latter has gathered significant interest in the NLP community recently [65, 66], focusing on language understanding, when grounded in an external context [67]. To a large extent, the focus of such studies has been on grounding language in a perceptual environment (e.g., image captioning [68–70], instruction following [71–73], and game playing [74, 75] tasks). Unlike these works, in this paper, we focus on *grounding language in a social context*, i.e., modeling the common ground [76–78] between the author and their social media followers, that enables understanding an otherwise highly ambiguous utterance. The *Social Context Understanding*, needed for building successful models for such tasks, can come from a wide variety of sources. The politician's affiliation and historical stances on the issue can capture crucial social context. Social relationships, knowledge about the involved entities, and related prior and upcoming events form important part of the puzzle as well. In fig. 3.1 event #1, combining the event information (*school shooting*) with the speakers' gun control stances, would facilitate understanding the *intended meaning* of the text.

The main motivation of this paper work is to operationalize the *'Social Context Grounding'* problem as a pragmatic understanding task. From a practical perspective, this would enable the creation of better NLP-CSS models that can process social media text in settings that require contextualized understanding. We suggest several datasets, designed to evaluate this ability in computational models. These task capture the intended meaning at different level of granularity. At the most basic level, providing the social context can help identify the entities targeted, and the sentiment towards them. In fig. 3.1, the social context ⟨event#1, Harris⟩and the text *"we need to keep our teachers safe"* ⇒ *"negative attitude towards guns"*. A more nuanced account of meaning, which we formulate as a separate task, captures the specific means in which the negative attitude is expressed (the Interpretation in fig. 3.1).

We additionally present two datasets corresponding to these tasks, namely, *'Target Entity and Sentiment Detection'* and *'Vague Text Disambiguation'*. In the first, the goal is to predict:

Tweet Target Entity and Sentiment				Vague Text Disambiguation
Tweet: As if we needed more evidence. #kavanaugh **Event:** Kavanaugh Supreme Court Nomination **Author:** Earl Blumenauer (Democrat Politician) **Targets:** Brett Kavanaugh (negative), Julie Swetnick (positive) Christine Ford (positive), Deborah Ramirez (positive)				**Vague Text:** First, but not the last. **Event:** US withdraws from Paris climate agreement that enforces environmental targets after three years **Author Party:** Republican **Disambiguation:** The withdrawal from the Paris climate agreement is the first step of many to come for the Trump administration. It will not be the last, as more positive changes are sure to follow. **Incorrect Disambiguations:**
Target Task Data Statistics		**Vague Text Data Statistics**		1) Joe Biden's inauguration marks the first day of a new era of progress and prosperity, lasting positive changes are coming. (Incorrect Event)
Unique Tweets	865	Unique Vague Texts	93	2) The Paris Climate Agreement withdrawal is the first of many backward steps this Trump administration is sure to take in destroying our environment. (Incorrect Stance)
Positive Targets	1513	Positive Examples	739	
Negative Targets	1085	Negative Examples	2217	
Neutral Targets	784	Total Examples	2956	
Non-Targets	2509	Number of Events	9	3) This is the time for America to move forward and make progress without being held back by a global agreement that doesn't serve our interests. (Doesn't match the vague text)
Total Data Examples	5891	Hard Test Examples	180	
Number of Events	3			

Table 3.1. Examples of Annotated Datasets and their statistics

1) whether a given entity is the *intended target* of a politician's tweet and 2) the sentiment towards the intended targets. We explicitly focus on tweets that *do not always mention the targets* in their text to incentivize modeling the pragmatic communicative intent of the text. In the second task, given an ambiguous political message such as *"We demand justice"* and its social context (*associated event*, & the *author's party affiliation*), the task is to identify a *plausible* unambiguous explanation of the message. Note that the ground truth for all these tasks is based on human pragmatic interpretation, i.e., "*guns*" is a negative target of "*we need to keep our teachers safe*", despite not being mentioned in the text, since it was perceived in this way by a team of human annotators reading the tweet and knowing social context. We show examples of each task in table 3.1. We describe the datasets in detail in section 3.1.

We evaluate the performance of various models, as a way to test the need for social context and compare different approaches for modeling it. These include pre-trained LM-based classifiers, and LLM in-context learning [14, 79], which use a textual representation of the social context. We also adopt an existing graph-based discourse contextualization framework [80, 81], to explicitly model the social context needed to solve the proposed tasks. Our results demonstrate that the discourse contextualization models outperform other models on both tasks. We present an error analysis to gain further insights. We describe the models in section 3.2 and the results in section 3.3.

We also present a qualitative visualization of a political event, *Brett Kavanaugh Supreme Court Nomination* (section 3.4.4), from target entity-sentiment perspective. It showcases a unique summary of the event discourse. We perform human evaluation on our *'Vague Text Disambiguation'* dataset, and observe that humans find this task much easier than the evaluated models. We also present observations of human vs. LLM errors in disambiguation. In summary, our contributions are:

1. Defining and operationalizing the *Social Context Grounding* task in political discourse
2. Evaluating various state-of-the-art context representation models on the task. We adopt existing discourse contextualization framework for the proposed tasks, and evaluate GPT-3's in-context learning performance, as well.
3. Performing human studies to benchmark the dataset difficulty and GPT-3 generation performance, when compared to human workers.[1]

3.1 Social Context Grounding Tasks

We design and collect two datasets for *Social Context Grounding* evaluation, and define three pragmatic interpretation tasks. In the *Tweet Target Entity and Sentiment* dataset, we collect annotations of opinionated tweets from known politicians for their intended targets and sentiments towards them. We focus on three political events for this task. The dataset and its collection are described below in section 3.1.1. In the *Vague Text Disambiguation Task*, we collect plausible explanations of vague texts, given the social context, consisting of *author affiliation* and *specific event*. We focus on eight political events. This dataset is detailed in section 3.1.2. Examples and data statistics are shown in table 3.1.

3.1.1 Tweet Target Entity and Sentiment Task

In this task, given a tweet T, its context, and an entity E, the objective is to predict whether or not E is a target of T and the sentiment towards E. Political discourse often contains opinionated discourse about world events and social issues. We collect tweets that don't directly mention the target entities. Thus, connecting the text with the event details

[1]↑Our data and code will be released under MIT license

and the author's general perspectives is necessary to solve this task effectively. We pick the focal entities for the given event and let human annotators expand on that initial set, based on their interpretation of the contextualized text. A *target* entity is conceptualized as an entity present in the full intended interpretation of the tweet.

We focus our tweet collection on three recent divisive events: *George Floyd Protests*, *2021 US Capitol Attacks*, and *Brett Kavanaugh's Supreme Court Nomination*. We identify relevant participating entities for each of the three events. Examples of the involved entities for the event *George Floyd Protests* were *George Floyd, United States Police, Derek Chauvin, Donald Trump, Joe Biden, United States Congress, Black people, Democratic Party, Republican Party, BLM, Antifa*.

Target-Sentiment Data Collection: We filter $3,454$ tweets for the *three* events using hashtags, keyword-based querying, and the dates of the event-based filtering from the Congress Tweets repository corpus[2]. We collect a subset of $1,779$ tweets that contain media (images/video) to increase the chances of the tweet text not containing the target entity mentions. Then, we use 6 in-house human annotators and Amazon Mechanical Turk (AMT) workers who are familiar with the event context for annotation. We ask them to annotate the targeted entities and sentiments towards the targets. The authors of this paper also participated in the annotation process. We provide them with entity options based on the event in the focus of the tweet. Annotators are allowed to add additional options if needed. We also ask the annotators to mark non-targets for each tweet. We instruct them to keep the non-targets as relevant to the event as possible to create harder negative examples. Each tweet is annotated by three annotators. We filter 865 unique tweets with $5,891$ annotations, with majority agreement on each tweet. All the AMT annotations were additionally verified by in-house annotators for correctness. AMT workers were paid USD 1 per tweet. It took 3 minutes on average for each assignment, resulting in an hourly pay of USD 20. We include screenshots of the collection task GUIs in the appendix. We split the train, and test sets by events, authors, and targets to incentivize testing the general social grounding capabilities of the models. The test set also consists of authors, targets, and events not seen in the training set. We use *Capitol Riots* event for the test set of *Target Entity and Sentiment* Task. We

[2]↑https://github.com/alexlitel/congresstweets

split the examples into 4,370 train, 511 development, and 1,009 test examples. We compute the mean Cohens kappa score for annotations and report inter-annotator agreement for annotated targets (0.47) and sentiment (0.73)

3.1.2 Vague Text Disambiguation Task

The task of *Vague Text Disambiguation* is designed to capture pragmatic interpretation at a finer-grained level. It can be viewed as a variant of the well known paraphrase task, adapted for the social context settings. The model is evaluated on its ability to identify plausible interpretations (i.e., a sentence explicitly describing the author's intent) of an ambiguous quote given the event context and author's affiliation. E.g., "*protect our children from mass shootings*" could easily be disambiguated as either "*ban guns*" or "*arm teachers*" when the author's stance on the issue of '*gun rights*' is known.

Our data collection effort is designed to capture different aspects of social context grounding and facilitate detailed error analysis. Defined as a binary classification task over tuples ⟨`Party, Event, Vague text, Explicit text`⟩, we create negative examples by flipping tuple elements values of positive examples. This allows us to evaluate whether models can capture event relevance, political stance, or constrain the interpretation based on the vague text. For example, in the context of Event #1 in fig. 3.1, we can test if models simply capture the correlation between Democrats and negative stance towards guns access by replacing the vague text to *"let your voice be heard"*, which would make the interpretation in fig. 3.1 implausible despite being consistent with that stance, while other consistent interpretations would be plausible (e.g., *"go outside and join the march for our lives"*).

Vague Text Data Collection: Data collection was done in several steps. (1)**Vague Texts Collection**. We collected vague text candidates from tweets by US politicians (i.e. senators and representatives) between the years 2019 to 2021 from Congress Tweets corpus. We identified a list of 9 well-known events from that period and identified event-related tweets using their time frame and relevant hashtags. We used a pre-trained BERT-based [82] NER model to collect tweets that do not contain any entity mentions to identify potential

candidates for vague texts. We manually identified examples that could have contrasting senses by flipping their social context. We obtain 93 vague text candidates via this process. (2)**In-Context Plausible Meaning Annotation**. We match the 93 ambiguous tweets with different events that fit them. We use both Democrat and Republican as the author party affiliation. We obtain 600 *context-tweet* pairs for AMT annotation. For each tweet, we ask AMT workers to annotate the following two aspects: 1) sentiment towards the three most relevant entities in the event (sanity check) and 2) a detailed explanation of the *intended meaning* given the event and author's party affiliation. We obtain 469 reasonable annotations. After this step, each annotation was screened by in-house annotators. We ask three in-house annotators to vote on the *correctness*, *appropriateness*, and *plausibility* of the annotation given the context. Thus, we create a total of 374 examples.
(3)**LLM-based Data Expansion**. Using these examples, we further generate candidates for the task using LLM few-shot prompting. We use the examples from the previous step as in-context few-shot examples in the prompt. We use GPT-NeoX [79] and GPT-3 [14] for candidate generation. For each generated answer, manual inspection by three in-house annotators is performed to ensure data quality. We generate 928 candidates using GPT-NeoX and GPT-3. Human expert filtering results in 650 generations that pass the quality check. After removing redundant samples, we obtain 365 examples. Thus, we obtain a total of 739 annotations for this task. Then, for each of the 739 examples, we ask in-house annotators to select 3 relevant negative options from the pool of explanations. We instruct them to pick hard examples that might contain similar entities as the correct interpretation. This results in $2,956$ binary classification data samples. We analyze and discuss the results of human validation of large LM generations in section 3.4.

Similar to the previous task, we split the train, test sets by events, and vague text to test the general social understanding capabilities of the model. We reserve *Donald Trump's second impeachment verdict* event for the test set. We also reserve Democratic examples of 2 events and Republican examples of 2 events exclusively for the test set. We split the dataset into $1,916$ train, 460 development, and 580 test examples. 180 of the test examples are from events/party contexts unseen in train data.

3.2 Modeling Social Context

Model		Target Identification				Sentiment Identification			
		Prec	Rec	Macro-F1	Acc	Prec	Rec	Macro-F1	Acc
No Context Baselines	BERT-large	69.09	72.35	68.83	70.56	58.74	60.17	58.95	58.37
	RoBERTa-base	66.58	69.54	65.14	66.40	61.68	61.27	61.36	60.65
PLMs +Twitter Bio Context	BERT-large + user-bio	69.03	71.86	69.34	71.66	60.02	60.44	60.13	59.86
	RoBERTa-base + user-bio	65.83	68.65	64.79	66.30	60.06	59.91	59.94	59.46
PLMs +Wikipedia Context	BERT-large + wiki	63.58	65.78	60.33	61.05	53.48	56.44	53.9	53.32
	RoBERTa-base + wiki	69.02	72.32	68.62	70.27	57.62	59.10	58.07	58.28
LLMs	GPT-3 0-shot	69.25	70.58	69.77	73.78	56.20	55.04	54.18	56.80
	GPT-3 4-shot	69.81	72.99	66.45	67.03	58.12	57.10	55.00	57.51
Static Contextutalized Embedding Models	RoBERTa-base + PAR Embs	68.38	71.63	67.67	69.18	55.01	56.89	55.51	55.40
	BERT-large + PAR Embs	65.40	67.33	60.25	60.56	55.24	57.54	55.89	55.80
	RoBERTa-base + DCF Embs	**72.89**	**75.95**	**73.56**	**75.82**	63.05	63.52	62.90	63.03
	BERT-large + DCF Embs	68.76	72.02	68.32	69.97	61.59	63.25	61.22	60.75
Discourse Contextualized Models	BERT-large + DCF	71.12	74.61	71.17	72.94	**65.81**	**65.25**	**65.34**	**65.31**
	RoBERTa-base + DCF	70.44	73.86	70.39	72.15	63.45	63.34	63.37	63.23

Table 3.2. Results of baseline experiments on *Target Entity* (binary task) and *Sentiment* (4-classes) test sets. We report macro-averaged Precision, macro-averaged Recall, macro-averaged F1, and Accuracy metrics.

The key technical question this paper puts forward is how to model the social context, such that the above tasks can be solved with high accuracy. We observe that humans can perform this task well (section 3.4.3), and evaluate different context modeling approaches in terms of their ability to replicate human judgments. These correspond to `No Context`, `Text-based` context representation (e.g., Twitter Bio, relevant Wikipedia articles), and `Graph-based` context representation, simulating the social media information that human users are exposed to when reading the vague texts.

We report the results of all our baseline experiments in table 3.2 and table 3.3. The first set of results evaluate fine-tuned pre-trained language models (PLM), namely BERT [82] and RoBERTa [83], with three stages of modeling context. Firstly, we evaluate no contextual information setting. Second, we include the authors' Twitter bios as context. Finally, we evaluate the information from the author, event, and target entity Wikipedia pages as context (models denoted `PLM Baselines {No, Twitter Bio, Wikipedia} Context`, respectively).

We evaluate GPT-3[3] in *zero-shot* and *four-shot* in-context learning paradigm on both tasks. We provide contextual information in the prompt as short event descriptions and

[3]↑gpt-3.5-turbo-1106 via OpenAI API

Model	Vague Text Disambiguation			
	Prec	Rec	Macro-F1	Acc
No Context Baselines				
BERT-large	52.24	55.58	50.28	53.75
RoBERTa-base	55.3	51.82	54.53	56.08
PLMs + Wikipedia Context				
BERT-large + wiki	52.31	46.90	66.87	76.03
BERT-base + wiki	51.85	38.62	64.36	75.69
LLMs				
GPT-3 0-shot	**63.10**	62.92	62.58	63.5
GPT-3 4-shot	62.05	62.29	61.86	62.04
Static Contextutalized Embedding Models				
BERT-large + PAR	47.68	49.66	65.53	73.79
BERT-base + PAR	45.93	54.48	65.49	72.59
BERT-large + DCF Embs	47.18	**63.45**	67.55	73.10
BERT-base + DCF Embs	56.58	59.31	**71.71**	**78.45**
Discourse Contextualization Models				
BERT-large + DCF	52.76	59.31	69.94	76.55
BERT-base + DCF	52.73	60.00	70.06	76.55

Table 3.3. Results of baseline experiments on *Vague Text Disambiguation* dataset test split, a binary classification task. We report macro-averaged Precision, macro-averaged Recall, macro-averaged F1, and Acc. metrics

authors' affiliation descriptions. Note that GPT-3 is trained on news data until Sep. 2021 which includes the events in our data (models denoted `LLM Baseline`).

We evaluate the performance of politician embeddings from Political Actor Representation (PAR) [81] and Discourse Contextualization Framework (DCF) [80] models. (models denoted `Static Contextutalized Embeddings`). We use PAR embeddings available on their GitHub repository[4]. For DCF model, we use released pre-trained models from GitHub repository[5] to generate author, event, text, and target entity embeddings. We evaluate the embeddings on both tasks. We briefly review these models in section 3.2.1 & section 3.2.2.

Finally, we use tweets of politicians from related previous events and build context graphs for each data example as proposed in [80]. We use Wikipedia pages of authors, events, and target entities to add social context information to the graph. Then, we train the Discourse Contextualization Framework (DCF) for each task and evaluate its performance

[4]↑https://github.com/BunsenFeng/PAR
[5]↑https://github.com/pujari-rajkumar/compositional_learner

on both tasks (models denoted `Discourse Contextualization Model`). Further details of our baseline experiments are presented in subsection section 3.2.3. Results of our baseline experiments are discussed in section 3.3.

3.2.1 Discourse Contextualization Framework

Discourse Contextualization Framework (DCF) [80] leverages relations among social context components to learn contextualized representations for text, politicians, events, and issues. It consists of *encoder* and *composer* modules that compute holistic representations of the context graph. The encoder creates an initial representation of nodes. Composer propagates the information within the graph to update node representations. They define link prediction learning tasks over context graphs to train the model. They show that their representations significantly outperform several PLM-based baselines trained using the same learning tasks.

3.2.2 Political Actor Representation

[81] propose the *Political Actor Representation* (PAR) framework, a graph-based approach to learn more effective politician embeddings. They propose three learning tasks, namely, 1) Expert Knowledge Alignment 2) Stance Consistency training & 3) Echo chamber simulation, to infuse social context into the politician representations. They show that PAR representations outperform SOTA models on *Roll Call Vote Prediction* and *Political Perspective Detection*.

3.2.3 Experimental Setup

Target Entity Detection is binary classification with ⟨*author, event, tweet, target-entity*⟩as input and *target/non-target* label as output. *Sentiment Detection* is set up as 4-way classification. Input is the same as the target task and output is one of: {*positive, neutral, negative, non-target*}. *Vague Text Disambiguation* is a binary classification task with ⟨*party-affiliation, event, vague-text, explanation-text*⟩and a *match/no-match* label as output.

In phase 1 no-context baselines, we use the author, event, tweet, and target embeddings generated by PLMs. We concatenate them for input. In Twitter-bio models, we use the author's Twitter bio embeddings to represent them. Wiki context models receive Wikipedia page embeddings of author, event, and target embeddings. *It is interesting to note that the Wikipedia context models get all the information needed to solve the tasks.*. In phase 2 LLM experiments, we use train samples as in-context demonstrations. We provide task and event descriptions in the prompt. In phase 3 PAR models, we use politician embeddings released on the PAR GitHub repository to represent authors. We replace missing authors with their wiki embeddings. For the *Vague Text* task, we average PAR embeddings for all politicians of the party to obtain party embeddings. For DCF embedding models, we generate representations for all the inputs using context graphs. We also use authors' tweets from relevant past events. We build graphs using author, event, tweet, relevant tweets, and target entity as nodes and edges as defined in the original DCF paper. In phase 4, we use the same setup as the DCF embedding model and additionally back-propagate to DCF parameters. This allows us to fine-tune the DCF context graph representation for our tasks.

3.3 Results

The results of our baseline experiments are described in Tab. 3.2 and 3.3. We evaluate our models using macro-averaged precision, recall, F1, and accuracy metrics (due to class imbalance, we focus on macro-F1). Several patterns, consistent across all tasks, emerge. **First**, *modeling social context is still an open problem.* None of our models were able to perform close to human level. **Second**, *adding context can help performance*, compared to the No-Context baselines, models incorporating context performed better, with very few exceptions. **Third**, *LLMs are not the panacea for social-context pragmatic tasks.* Despite having access to a textual context representation as part of the prompt, and having access to relevant event-related documents during their training phase, these models under-perform compared to much simpler models that were fine-tuned for this task. **Finally**, *explicit context modeling using the DCF model consistently leads to the best performance.* The DCF model mainly represents the social context in the form of text documents for all nodes. Further symbolic

addition of other types of context such as social relationships among politicians and relationships between various nodes could further help in achieving better performance on these tasks. In the *Target Entity* task, RoBERTa-base + DCF embeddings obtain 73.56 F1 vs. 68.83 for the best no-context baseline. Twitter bio and wiki-context hardly improve, demonstrating the effectiveness of modeling contextual information explicitly vs. concatenating context as text documents. No context performance well above the random performance of 50 F1 indicates the bias in the target entity distribution among classes. We discuss this in section 3.4.4. In *Sentiment Identification* task, we see that BERT-large + DCF back-propagation outperforms all other models. *Vague Text Disambiguation* task results in table 3.3 show that DCF models outperform other models significantly. 71.71 F1 is obtained by BERT-base + DCF embeddings. BERT-base performing better than bigger PLMs might be due to DCF model's learning tasks being trained using BERT-base embeddings.

3.4 Analysis and Discussion

Democrat Only Entities		Common Entities					Republican Only Entities	
Target	Sentiment	Agreed-Upon Entities		Divisive Entities			Target	Sentiment
		Target	Sentiment	Sentiment (D)	Target	Sentiment (R)		
				Positive	Christine Blasey Ford	Negative		
Anita Hill	Positive	US Supreme Court	Neutral	Positive	Deborah Ramirez	Negative	Susan Collins	Positive
Patty Murray	Positive	US Senate	Neutral	Positive	Julie Swetnick	Negative	Chuck Grassley	Positive
Merrick Garland	Positive	FBI	Neutral	Negative	Brett Kavanaugh	Positive	Diane Feinstein	Negative
Jeff Flake	Negative	Judiciary Committee	Neutral	Negative	Donald Trump	Positive	Chuck Schumer	Negative
				Negative	Mitch McConnell	Positive	Sean Hannity	Neutral

Table 3.4. Target Entity-Sentiment centric view of *Kavanaugh Supreme Court Nomination* discourse

3.4.1 Ablation Analysis on Vague Text Task

We report ablation studies in table 3.5 on the Vague Text task test set. We consider 5 splits: **(1)** Unseen Party: ⟨*party, event*⟩not in the train set but ⟨*opposing-party, event*⟩is present, **(2)** Unseen Event: ⟨*party* ⟩not in train set, **(3)** Flip Event: negative samples with corresponding 'event flipped-party/vague tweet matched' positive samples in train set and analogous **(4)** Flip Party and **(5)** Flip Tweet splits. We observe the best model in each category. They obtain weaker performance on unseen splits, as expected, unseen events being the hardest. Contextualized models achieve higher margins. DCF gains 7.6(13.2%)

and DCF embeddings attain 8.12(20.42%) macro-F1 improvement over BERT-base+wiki compared to respective margins of 8.86% and 11.42% on the full test set. In the flip splits with only negative examples, accuracy gain over random baseline for all splits is seen. This indicates that models learn to jointly condition on context information rather than learn spurious correlations over particular aspects of the context. Specifically, flip-tweet split results indicate that models don't just learn party-explanation mapping.

Data Split	Unseen Party	Unseen Event	Flip Tweet	Flip Event	Flip Party
	Ma-F1	Ma-F1	Acc	Acc	Acc
Random	44.70	29.69	75	75	75
BERT-base+wiki	57.58	39.76	88.14	89.77	87.77
BERT-base +DCF Embs	61.79	47.88	86.10	93.18	84.57
BERT-base+DCF	65.18	45.65	82.03	89.77	84.04

Table 3.5. Ablation Study Results on Vague Text Task

3.4.2 Vague Text LLM Generation Quality

We look into the quality of our LLM-generated disambiguation texts. While GPT-NeoX [79] produced only 98 good examples out of the 498 generated instances with the rest being redundant, GPT-3 [14] performed much better. Among the 430 generated instances, 315 were annotated as good which converts to an acceptance rate of 20.04% for GPT-NeoX and 73.26% for GPT-3 respectively. In-house annotators evaluated the quality of the generated responses for how well they aligned with the contextual information. They rejected examples that were either too vague, align with the wrong ideology, or were irrelevant. In the prompt, we condition the input examples in all the few shots to the same event and affiliation as the input vague text. In comparison, the validation of AMT annotations for the same task yielded 79.8% good examples even after extensive training and qualification tests. Most of the rejections from AMT were attributed to careless annotations.

3.4.3 Vague Text Human Performance

We look into how humans perform on the *Vague Text Disambiguation* task. We randomly sample 97 questions and ask annotators to answer them as multiple-choice questions. Each vague text-context pair was given 4 choices out of which only one was correct. We provide a brief event description along with all the metadata available to the annotator. Each question was answered by 3 annotators. Among the 97 answered questions, the accuracy was 94.85%, which shows this task is easy for humans who understand the context. Respective performance of best models on this subset of data for BERT-base+wiki (54.89%), BERT-base+DCF-embs (63.38%), BERT-base+DCF (64.79%) is much lower than human performance.

3.4.4 Target Entity Visualization

The main goal of this analysis is to demonstrate the usefulness and inspire modeling research in the direction of entity-sentiment-centric view of political events. table 3.4 visualizes one component of how partisan discourse is structured in these events. We study *Kavanaugh Supreme Court Nomination*. We identify discussed entities and separate them into divisive and agreed-upon entities. This analysis paints an accurate picture of the discussed event. We observe that the main entities of Trump, Dr. Ford, Kavanaugh, Sen. McConnell, and other accusers/survivors emerge as divisive entities. Entities such as Susan Collins and Anita Hill who were vocal mouthpieces of the respective party stances but didn't directly participate in the event emerge as partisan entities. Supreme Court, FBI, and other entities occur but only as neutral entities.

3.4.5 DCF Context Understanding

We look into examples that are incorrectly predicted using Wikipedia pages but correctly predicted by the DCF model in the appendix (table 3.6). In examples 1 & 2 of *Target Entity-Sentiment* task, when the entity is not explicitly mentioned in the tweet, the Wiki-Context model fails to identify them as the targets. We posit that while the Wikipedia page of

each relevant event will contain these names, explicit modeling of entities in the DCF model allows correct classification. Examples $1-3$ of *Vague Text Disambiguation* task show that when no clear terms indicate the sentiment towards a view, the Wiki-Context model fails to disambiguate the tweet text. Explicit modeling of politician nodes seems to help the DCF model.

3.5 Conclusion and Future Work

In this paper, we motivate, define, and operationalize '*Social Context Grounding*' for political text. We build two novel datasets to evaluate social context grounding in NLP models that 'are easy for humans' when the relevant social context is provided. We experiment with many types of contextual models. We show that explicit modeling of social context outperforms other models while lacking behind humans.

3.6 Limitations

Our work only addresses English language text in US political domain. We also build upon large language models and large PLMs which are trained upon huge amounts of uncurated data. Although we employed human validation at each stage, biases could creep into the datasets. We also don't account for the completeness of our datasets as it is a pioneering work on a new problem. Social context is vast and could have a myriad of components. We only take a step in the direction of social context grounding in this work. The performance on these datasets might not indicate full social context understanding but they should help in sparking research in the direction of models that explicitly model such context. Although we tuned our prompts a lot, better prompts and evolving models might produce better results on the LLM baselines. Our qualitative analysis is predicated on a handful of examples. They are attempts to interpret the results of large neural models and hence don't carry as much confidence as our empirical observations. We believe the insights from our findings will encourage more research in this area. For example, the development of discourse contextualized models that aim to model human-style understanding of background knowledge, emotional intelligence, and societal context understanding is a natural next step of our research.

3.7 Ethics Statement

In this work, our data collection process consists of using both AMT and GPT-3. For the *Target Entity and Sentiment* task, we pay AMT workers $1 per HIT and expect an average work time of 3 minutes. This translates to an hourly rate of $20 which is above the federal minimum wage. For the *Vague Text Disambiguation* task, we pay AMT workers $1.10 per HIT and expect an average work time of 3 minutes. This translated to an hourly rate of $22.

We recognize collecting political views from AMT and GPT-3 may come with bias or explicit results and employ expert gatekeepers to filter out unqualified workers and remove explicit results from the dataset. Domain experts used for annotation are chosen to ensure that they are fully familiar with the events in focus. Domain experts were provided with the context related to the events via their Wikipedia pages, background on the general issue in focus, fully contextualized quotes, and authors historical discourse obtained from ontheissues.org. We have an annotation quid-pro-quo system in our lab which allows us to have a network of in-house annotators. In-house domain experts are researchers in the CSS area with familiarity with a range of issues and stances in the US political scene. They are given the information necessary to understand the events in focus in the form of Wikipedia articles, quotes from the politicians in focus obtained from ontheissues.org, and news articles related to the event. We make the annotation process as unambiguous as possible. In our annotation exercise, we ask the annotators to mark only high-confidence annotations that can be clearly explained. We use a majority vote from 3 annotators to validate the annotations for the target entity task.

Our task is aimed at understanding and grounding polarized text in its intended meaning. We take examples where the intended meaning is clearly backed by several existing real-world quotes. We do not manufacture the meaning to the vague statements, we only write down unambiguous explanations where context clearly dictates the provided meaning. Applications of our research as we envision would be adding necessary context to short texts by being able to identify past discourse from the authors that are relevant to the particular

text in its context. It would also be able to ground the text in news articles that expand upon the short texts to provide full context.

3.8 Reproducibility

We use the HuggingFace Transformers [84] library for PLMs. We use GPT-NeoX implementation by ElutherAI [79] and GPT-3 [85] via OpenAI API for our LLM baselines. We run 100 epochs for all experiments. We use 10 NVIDIA GeForce 1080i GPUs for our experiments. We use the train, development, and test splits detailed in section 3.1 for our experiments. We use the development macro-F1 for early stopping. We run all our experiments using random seeds to ensure reproducibility. We experiment with a random seed value set to {13}. We report the results of the 3 fold cross-validation. We report only the mean on all the cross-fold validation results for clarity. All our code, datasets, and result logs will be released publicly upon acceptance. We experiment with 3, 5, and 10 fold cross-validation. As the results on the development data are almost identical, we report the results of 3 fold cross-validation in all our experiments.

3.9 Annotation Interfaces

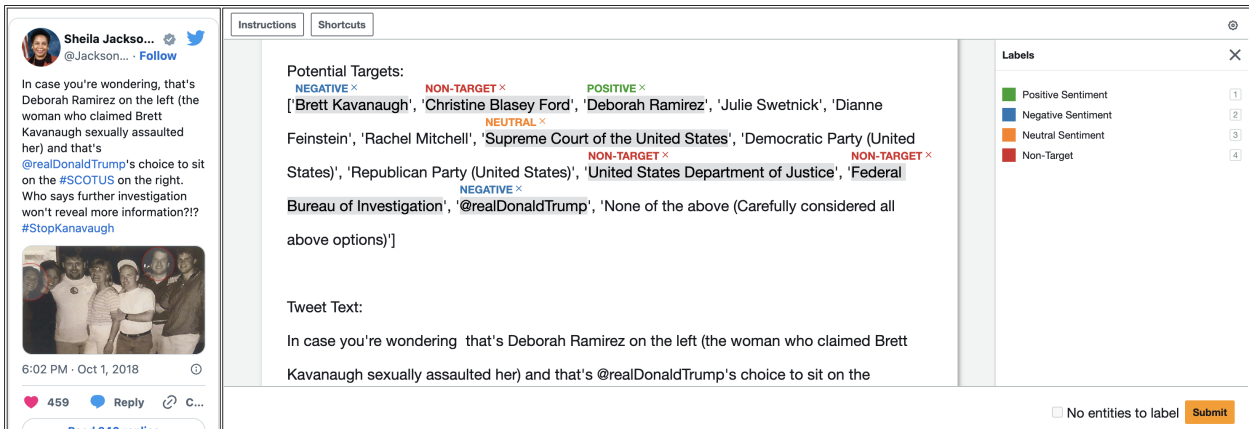


Figure 3.2. An example of *Tweet Target Entity and Sentiment Annotation* GUI

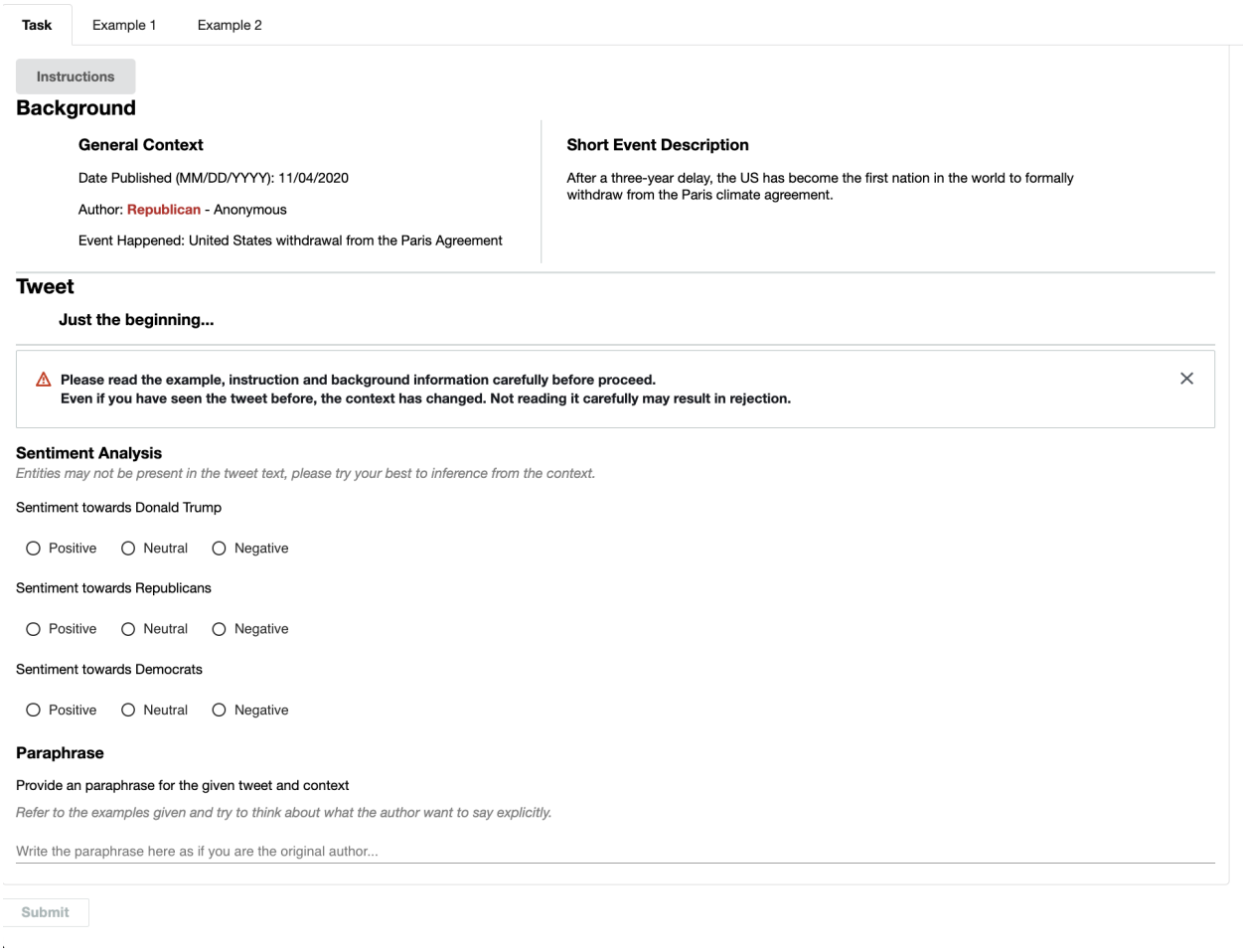


Figure 3.3. An example of *Vague Text Disambiguation* GUI

3.10 GPT Prompts

GPT-3 prompts used for target-entity task:

`Event: <event>`

`Event background: <background-description>`

`Tweet: <tweet-text>`

`Author: <author-name>`

`Author Party: <party-affiliation>`

`Author background: <first two sentences of author-wiki-page>`

`Target Entity: <entity-name>`

Entity background: <first two sentence of entity-wiki-page> Task: Identify if the given entity is a target of the tweet. A target entity is defined as an entity that would be present in the full unambiguous explanation of the tweet.
Is the given entity a target entity of the tweet? Answer yes or no.

GPT-3 prompts used for target-sentiment task:

Event: <event>
Event background: <background-description>
Tweet: <tweet-text>
Author: <author-name>
Author Party: <party-affiliation>
Author background: <first two sentences of author-wiki-page>
Target Entity: <entity-name>
Entity background: <first two sentence of entity-wiki-page> Task: Identify the sentiment of the tweet towards the given target entity. Consider that the tweet is ambiguous and the entity might be implied without being explicitly mentioned.
What is the sentiment of the tweet towards the target entity? Answer with positive, negative, or neutral.

GPT-3 prompts used for vague text disambiguation task:

Event: <event>
Event background: <background-description>
Vague message: <vague-text>
Author Party: <party-affiliation>
Author background: <first two sentences of party-wiki-page>
Task: Given the event, vague message, and party affiliation of the author, explain unambiguously the intended meaning of the vague message.
Generate an unambiguous explanation for the vague message given the party affiliation of the author and the event in context.

3.11 Error Analysis

Target Entity and Sentiment Task	Vague Text Disambiguation Task
Tweet: Republicans held Justice Scalias seat open for more than 400 days. Justice Kennedys seat has been vacant for less than two months. It's more important to investigate a serious allegation of sexual assault than to rush Kavanaugh onto the Supreme Court for a lifetime appointment. **Author**: Adam Schiff (Democrat) **Event**: Brett Kavanaugh Supreme Court nomination **Entity**: Christine Blasey Ford **Wiki-Context Prediction**: Not Target, **DCF Prediction**: Target (correct)	**Tweet**: Thanks for this. **Affiliation**: Democrat **Event**: United States withdrawal from the Paris Agreement **Paraphrase**: There's nothing surprising in withdrawing from the Paris agreement. Thanks for not caring our environment and future generations. **Wiki-Context Prediction**: No, **DCF Prediction**: Yes (correct)
Tweet: We will not be intimidated. Democracy will not be intimidated. We must hold the individuals responsible for the Jan. 6th attack on the U.S. Capitol responsible. Thank you @RepAOC for tonights Special Order Hour and we will continue our efforts to #HoldThemAllAccountable. **Author**: Adriano Espaillat (Democrat) **Event**: January 6 United States Capitol attack **Entity**: Donald Trump **Wiki-Context Predicted**: Not Target, **DCF Prediction**: Target (correct)	**Tweet**: Let us say enough. Enough. **Affiliation**: Democrat **Event**: Second impeachment of Donald Trump ended with not guilty **Paraphrase**: The failure of the Democrats to impeach Donald Trump is a strong moment for our legislature which can get back to its work helping the American people. Today we've been able to tell the American people what we have known all along, that Donald Trump was not guilty of these charges. **Wiki-Context Predicted**: Yes, **DCF Prediction**: No (correct)
Tweet: #GeorgeFloyd #BlackLivesMatter #justiceinpolicing QT @OmarJimenez Former Minneapolis police officer Derek Chauvin is in the process of being released from the Hennepin County correctional facility his attorney tells us. He is one of the four officers charged in the death of George Floyd. He faces murder and manslaughter charges. **Author**: Adriano Espaillat (Democrat) **Event**: George Floyd protests **Entity**: Derek Chauvin **Wiki-Context Predicted Sentiment**: Positive, **DCF Prediction**: Negative (correct)	**Tweet**: Lots of honking and screaming from balconies. Something must be going on. **Affiliation**: Democrat **Event**: Presidential election of 2020 **Paraphrase**: I'm sure that the people are celebrating the election results. **Wiki-Context Prediction**: No, **DCF Prediction**: Yes (correct)

Table 3.6. Examples where baseline model fails but DCF works

4. CULTURAL CONTEXT SCHEMA GROUNDING

Social norms define behavioral expectations shared across groups and societies [86]. They can help explain the differences in the way people from different cultural backgrounds react to the same situation [87, 88]. [89] describe *norms of interaction* as 'shared rules that implicate the belief system of a community', capturing the importance of representing norms when, either human or AI-systems, attempt to make sense of social interactions from different cultures.

Motivated by Large Language Models emergent reasoning abilities [16, 90–92] several recent works attempted to create repositories of cultural norms using novel prompting approaches followed by automated verification of the generated descriptions ([17–19]). However, these efforts had limitations such as using synthetic conversations or focusing on a handful of situations. LLMs also tend to suffer from several reliability issues such as hallucinations ([20]) or high sensitivity to prompt structure ([93–95]).

These limitations motivate a more general and principled approach for *Cultural Context Understanding*, which, we argue, should be viewed as a pragmatic reasoning task. Norms are situated in specific settings and are associated with the social roles participants play ([96]). Different expectations can be associated with individuals based on attributes such as their status, profession, or gender. Furthermore, these expectations are situation-dependent, for example, changing when engaging in professional or leisure activities.

To that end, in this paper, we propose a novel Cultural Context Grounding framework for conversations. We tackle three key problems. First, *norm representation* capturing the multi-party situated social expectations. Second, norm induction, i.e., populating the suggested representation with norm concepts, emerging from conversational data, and associating them with relevant contextual information. Finally, grounding the norm concepts in conversational data at scale and creating a dataset of norm-schema aligned conversations.

Our norm representation solution is inspired by the notion of *scripts* ([97]), i.e., structured representations of expected activities for different roles relevant to a specific scenario, in our case mapping to social scenarios. Unlike past work, [19], that developed a schema representing expectations over situated *actions*, our goal is to capture how social norms manifest in conversational behavior. To allow for pragmatic inferences mapping the norm

definition to conversations, we intentionally separate between *factual components*, capturing observed information about the settings and content of the conversation, and *cultural norm components*, capturing the expected behaviors and impact of violating that expectation. Fig. 4.1(a) depicts this separation.

Our framework follows a multi-stage approach to obtaining cultural information for existing conversations and grounding this information in conversation-specific details. We focus mainly on relevant social norms and their violations within these conversations. We then evaluate the usefulness of the created cultural context dataset both qualitatively and quantitatively. Fig. 4.2 presents an overview of our framework. We describe it in §4.3.

We leverage LLMs such as GPT-3.5 to generate a large but potentially noisy corpus of conversation-specific cultural information. Then, we leverage an *interactive human-in-the-loop* process (Alg. 1) that efficiently utilizes culturally proficient human annotation to organize this information into meaningful concepts. We further ground the generated descriptions in the conversation details, such as participants, their relationships, and their behavior using symbolic annotation. Following this, we experiment with automated verification strategies to filter the obtained cultural information at scale. Then, we organize the conversations and the obtained cultural information into a meaningful schema structure. Finally, we propose a neural graph schema model that leverages obtained data to improve the empirical performance on conversation understanding tasks such as emotion detection, sentiment detection, and dialogue act detection across multiple datasets. Overall, our contributions can be summarized as follows:

1. We propose a novel Cultural Grounding pipeline for conversation understanding using LLM & culturally-aware human annotation.
2. We introduce *Norm Concepts* and employ a human-in-the-loop (HiL) framework to create human-validated concepts supported by data.
3. We leverage automated verification strategies to clean the LLM-generated data. We further symbolically ground the conversations in norm concepts. We create a high-quality, large-scale dataset for cultural understanding.

4. We leverage the cultural schema information to improve downstream conversational task performance. We present meaningful visualizations and human evaluation experiments to showcase the quality.[1]

4.1 Related Work

Social Norm Datasets: A few recent works have leveraged LLMs such as GPT-3 to create useful cultural norm datasets using structure prompting approaches ([17–19]). While these works either focus on a small set or synthetically generated conversations, we generate $63k$ norm descriptions for $23.5k$ real conversations and ground them using a principled cultural grounding pipeline.

Social Grounding: [98] propose an interactive concept discovery for COVID-19 tweets. Several works address the tasks of principled grounding in social domain ([99–103]). However, they mainly focus on social grounding in political settings. We focus on the cultural aspect of social context which is challenging to obtain explicit data.

Automated Verification using LLMs: Several previous works address automated annotation using LLMs and refining their generations. ([104–109]). We build upon the frameworks proposed by [110] and [111] to operationalize our annotation framework.

4.2 Data

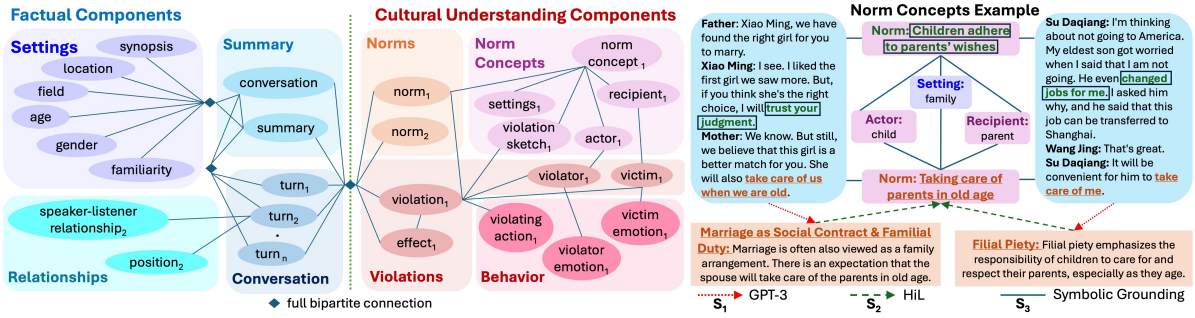
We build upon three existing dataset: MPDD ([112]), CPED ([113]), and LDC CCU Chinese Text[2] datasets. These datasets consist of Chinese conversations annotated with emotions, sentiments, dialogue acts, and social norm violations. Each dataset is annotated for some of these tasks, but not all. Detailed statistics of the dataset are presented in Tab.4.1. We describe each dataset briefly below.

MPDD Dataset: ([112]) This dataset contains 4,141 dialogues from Chinese TV series scripts. It is annotated with emotions, listeners, and speaker-listener relationships for each turn. The dataset contains 25,546 conversation turns.

[1]↑Code and Data: https://github.com/pujari-rajkumar/cultural-schema-naacl2025
[2]↑https://www.darpa.mil/program/computational-cultural-understanding

Dataset: MPDD	
Language: Chinese **Description:** Conversations from Chinese TV series scripts	**# Convs:** 4,141 **# Turns:** 25,546 **Tasks:** Emotion
Dataset: CPED	
Language: Chinese **Description:** Textual dataset with multi-modal features from 40 Chinese TV shows	**# Convs:** 11,832 **# Turns:** 132,723 **Tasks:** Dialogue Act, Emotion, & Sentiment
Dataset: LDC CCU ZH Text	
Language: Chinese **Description:** Chinese textual dataset containing online forum and chat conversations	**# Convs:** 6,763 **# Turns:** 98,821 **Tasks:** Norm Violation, Emotion, & Dialogue Act

Table 4.1. List of our raw data sources. ZH: Chinese; #Convs: Conversations; #Turns: Conversation Turns



(a) Proposed Schema Structure

(b) *Norm Concepts* & *Symbolic Explanations* example. S_* denotes relevant pipeline stages from Fig. 4.2.

Figure 4.1. Proposed *Cultural Context Schema Structure for Conversations* with an example instance

CPED Dataset: ([113]) CPED contains transcripts from 40 Chinese TV shows. It contains 11,832 conversations. It is annotated for emotions, sentiments, and dialogue acts. This dataset also provides multi-modal features such as face position, etc. We don't use them in this work.

LDC CCU ZH Text Dataset: This dataset is a conversational understanding dataset from the DARPA CCU program. This dataset consists of text, audio, and video data in several languages. We use the text portion of the Chinese dataset for our experiments. It contains annotations for emotion, dialogue act, social norm violation status, and conversation

change points. We focus on emotion and dialogue act identification tasks. This dataset also has metadata about settings, age of participants, familiarity, etc, annotated. The dialogue act annotation for this dataset is not complete. The annotation is focused only on a pre-determined set of dialogue acts. We deal with this by using *other* class for unlabeled data.

As not all datasets have all components annotated, we train the Llama-3.1-70b-Instruct ([114]) model to predict missing fields for each dataset using QLoRA ([115]) fine-tuning. We predict relationships for CPED and LDC datasets. We predict LDC-style metadata for CPED and MPDD datasets.

4.3 Cultural Context Grounding

Our high-level objective is to '*conceptualize & operationalize cultural understanding in conversational behavior*.' We identify three key steps to make progress towards this objective:

1. Formalize relevant cultural context for better conversational understanding.
2. Obtain high-quality cultural context data at scale, leveraging native-culture expertise and ground the conversations in this context.
3. Evaluate the obtained cultural context dataset for *correctness* and *usefulness*.

First, we propose a graph-based schema structure for culturally enriched conversational representation. Our schema consists of two complementing segments: *factual* components and *cultural understanding* components. We present an overview of the schema structure in Fig. 4.1 and a detailed description of the schema structure in §4.3.1.

Then, we introduce a robust pipeline for obtaining cultural information for real conversations and grounding the conversations in it. We efficiently leverage (1) native-culture human expertise, (2) LLMs as knowledge elicitors, (3) LLMs as symbolic annotators, and (4) LLMs as multi-agent verifiers, in this pipeline. We obtain a large-scale corpus for ∼23k Chinese conversations from three existing datasets (Tab. 4.1). We present an overview of the proposed pipeline in Fig. 4.2 and a detailed pipeline description in §4.3.2.

Finally, we evaluate (1) *correctness* of the cultural information obtained using elicitor & symbolic annotator LLMs, and (2) *usefulness* of the schema for conversational understanding. We measure *correctness* against human annotations (§4.4). We evaluate the *usefulness* of

cultural schema data via conversational tasks such as emotion, sentiment, and dialogue act detection (§4.5). We present full statistics of our pipeline data collection process in Tab. 4.7 in appendix 4.18.

4.3.1 Schema Structure

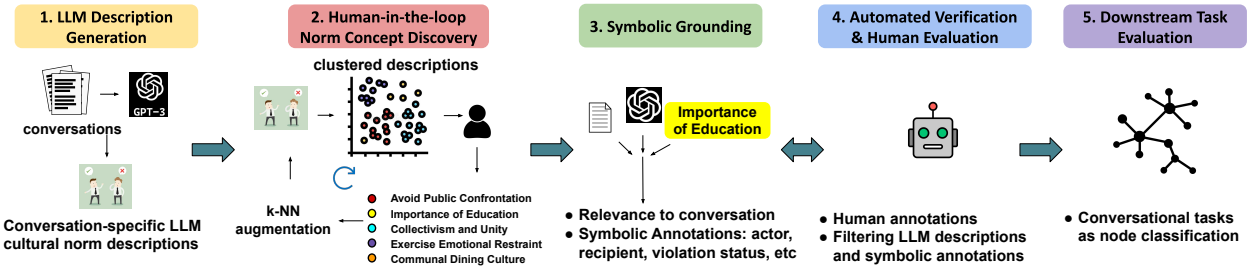


Figure 4.2. Proposed Cultural Context Grounding Pipeline for Conversations

Context often dictates whether or not a specific behavior is considered normative. For example, while patronizing younger people might be commonplace in families in some cultures, it could be considered offensive in a professional setting. Especially if the younger person is a work superior. On the other hand, even within the same social setting, norms might vary circumstantially. While joking about an elder's forgetful nature might be okay in some situations, it could be considered insensitive if they are suffering from dementia.

This guides us toward two distinct genres of information that could influence conversational behavior: factual and cultural information. To account for this, we propose a schema structure with two distinct segments (Fig. 4.1(a)). The *factual* segment of our schema consists of settings, summary, conversation, & relationship components. *Cultural Understanding* segment consists of social norms, violations, norm concepts, & behavior components.

While factual information such as age group, location, and relationships is available for many datasets, descriptions of relevant social norms, their violations, and how they affect the conversation are harder to procure. We address this challenge by efficiently using LLMs and human annotation.

4.3.2 Grounding Pipeline

Grounding in NLP usually refers to linking text or speech to real-world concepts such as entities, attitudes, etc ([116]). Conversational grounding can encompass various aspects, such as mutual beliefs, shared knowledge, assumptions, and so on ([117]). In this work, we focus specifically on the *cultural knowledge* aspect of conversational understanding.

Obtaining exhaustive knowledge of a culture's social norms at scale is impractical. Therefore, we propose a bottom-up approach. We use real-world conversations to mine situation-dependent social norms, leveraging LLMs as *knowledge elicitors*. Native-culture human annotators then create structured abstractions over these descriptions, which we refer to as *norm concepts*. Building upon [98], we devise an interactive framework that amplifies human judgments and scales them to the entire dataset. Then, we use LLMs as *symbolic annotators* to ground conversations in human-generated norm concept structures.

As LLMs are susceptible to hallucinations and bias, we evaluate the obtained dataset and symbolic annotations against human judgments. Then, we further employ LLMs as multi-agent verifiers to significantly improve the quality of the dataset. We present an overview of the proposed pipeline approach in Fig. 4.2. We further discuss details of each step of the pipeline in the following subsections.

LLM Description Generation

Two major challenges we face in building our cultural context dataset are (1) collecting structured knowledge about social norms as they manifest in real-world conversations and (2) scaling the descriptions of culturally nuanced behaviors observed in conversations. We address these challenges by leveraging existing conversational datasets and LLMs as *knowledge elicitors*, respectively. We provide raw conversations to the LLM and instruct it to explain observed behavior from a cultural perspective. For each conversation, we obtain descriptions of relevant social norms, potential violations, and their effects on the conversation trajectory and the participants. We present an example of norm descriptions in Fig. 4.1(b). We further provide the exact prompts and a full example of LLM outputs in the appendix 4.15. We expect this step to be noisy as LLMs are susceptible to hallucinations. However,

our primary focus is to obtain broad coverage of diverse cultural nuances that influence behavior in conversations. We evaluate the quality of LLM generations in §4.4.

HiL Norm Concept Discovery

As noted in [117], humans actively draw from a *'common ground'* when engaging in conversations. From a cultural standpoint, this common ground includes shared cultural beliefs. Humans often encounter social interactions where cultural awareness is practiced, which makes them adept at *situating* conversations in *cultural common ground*. Many aspects of this cultural common ground are highly general and serve us in a variety of situations. However, structured datasets that serve as a common ground for NLP models are hard to create solely using either human annotation or automated methods. Hence, in an attempt to create such a resource at scale, we employ culturally aware humans in an interactive human-in-the-loop (HiL) framework, which amplifies their judgments in a semi-automatic fashion.

Algorithm 1 Interactive Norm Concept Discovery

Input: Conversations and their norm descriptions
Outputs: *Norm Concepts*, *Symbols*, many-to-one mapping: ⟨description, conv⟩ → concept

1: Cluster norm descriptions using k-means
2: Cultural experts create *norm concepts* by selecting 5-10 samples of closely related descriptions and providing symbolic structure description for the *norm concept*
3: Perform k-NN augmentation of unmapped norm descriptions to each norm concept
4: Experts inspect augmented samples and mark 5-10 good, bad samples for each concept
5: Re-assign norm descriptions to concepts using good & bad cluster centers
6: Go back to step 1 with the remaining unmapped norm descriptions

Among LLM-generated cultural data, we observe overlapping norm descriptions across conversations, with minor variations. More interestingly, we also find closely related descriptions that can be grouped under the same theme. Consider the example in Fig. 4.1(b). Norm descriptions '*marriage as a familial duty: spouse is expected to take care of parents in old age*' and '*filial piety towards parents in old age*' are both defined by the common theme '*care of parents in old age as child's responsibility.*'

Concept Name: Respect For Authority
Description: Respecting hierarchies in family, professional, & organizational settings. It involves individuals respecting the decisions, suggestions, orders, and advice from those in higher positions
Settings: workplace, family, organizations
Violation Sketch: Behavior that intentionally contradicts the expectations and decisions of the people in charge
Actors: sub-ordinates or people in an inferior social position such as students, children, etc
Recipients: people in a position of power or authority over other people

Table 4.2. Cultural expert annotation of symbolic structure for discovered norm concept *Respect for Authority*

To capture such themes in a principled manner, we introduce ***norm concepts***. They are abstractions over cultural beliefs which influence several related behaviors. We associate them with symbolic explanations, thus creating structured representations.

A norm concept is characterized by an activation setting, a violation sketch, and actor and recipient roles. *Actor* role describes people expected to adhere to the norm concept. *Recipients* perceive/experience the consequences of adherence or violation. An example concept, *'Respect for Authority'*, is presented in Tab. 4.2. The symbolic structure of norm concepts is shown in Fig.4.1(b). The goal of HiL methodology is to support the discovery process via interaction with data.

[98] propose an interactive concept learning framework for tweets. We extend their framework for norm concept discovery. We outline our process in Alg. 1 and describe it below.

Norm concepts are *validated by humans* and *supported by data*. We create initial unnamed clusters of LLM norm descriptions. Humans inspect these clusters and create norm concepts by selecting 5-10 closely related examples of the concept. Humans also define a symbolic structure for the norm concept. Then, we perform k-NN augmentation of the concepts using untouched descriptions.

Then, humans inspect the newly augmented samples for each norm concept and mark 5-10 good and bad examples for each concept cluster. We re-perform k-NN augmentation for the untouched descriptions. Humans also create new norm concepts when they deem appropriate. We perform further iterations to discover more norm concepts. As the process progresses, the unnamed clusters evolve and reveal new concepts. This iterative process helps us reliably amplify human mapping decisions to the entire dataset. We discovered 35 norm concepts during this phase with a coverage of 64% over $67k$ norm descriptions we collected. Annotation interface screenshots are presented in appendix 4.19. We evaluate the results of the concept discovery process using human evaluation in §4.4. All annotators are CSS graduate students. We discuss this in detail in appendix 4.13.

Symbolic Grounding

The HiL process creates norm concepts and maps them to conversation-specific descriptions. However, to fully ground the conversation in a norm concept, we must also align the conversation to the concept's symbolic structure. Hence, we leverage LLMs as *symbolic annotators* to identify instantiations of concept symbols in the conversation.

We provide the symbolic annotator LLM with a conversation, LLM-generated descriptions, and associated norm concept structure. We first ask to verify the *relevance* of the description to the conversation (filtering hallucinations) and the *correctness* of description-concept mapping (filtering incorrect HiL mappings). Then, we ask it to annotate violation status, actor roles, recipient roles, and violation details, if applicable. We provide the exact prompts and outputs in appendix 4.16. We evaluate violation status annotations against human annotation in §4.4.

Automated Verification

As we use LLMs and semi-automated decision amplification, the dataset is prone to noise from hallucinations and incorrect mappings. Hence, we devise techniques to refine the dataset in a systematic manner. [18] have shown that self-verification improves the LLMs output quality significantly. We aim to further improve this process by leveraging LLMs

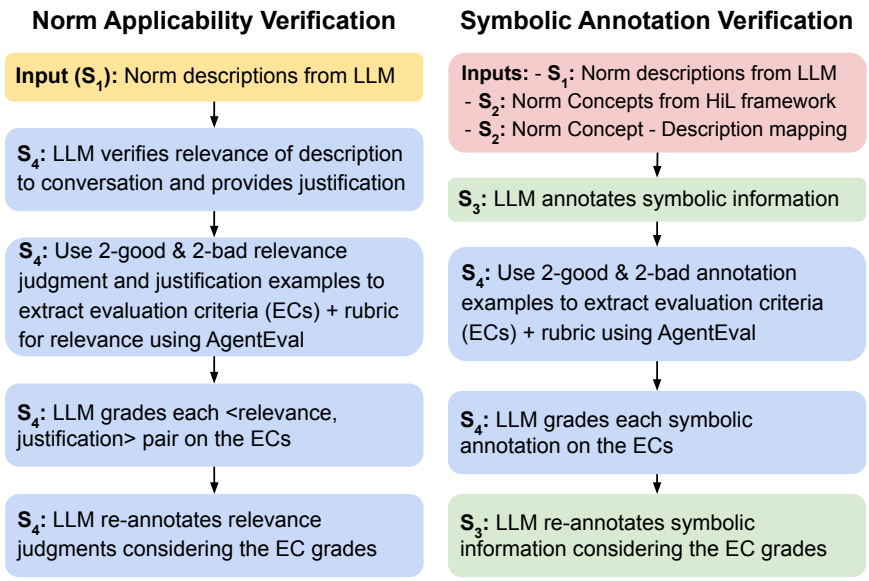


Figure 4.3. Automated Verification Flowcharts. S_* denotes pipeline stages from Fig. 4.2

as multi-agent verifiers. To achieve this, we employ *AgentEval* tool proposed by [118]. We evaluate the initial, self-verified, and AgentEval-refined datasets against human judgments in §4.4. We refine both the LLM descriptions and symbolic grounding annotations using AgentEval.

Self-Verification is the technique of prompting an LLM to re-consider a judgment. This is shown to be an effective technique in improving LLM response quality ([18, 119]). We employ this technique to refine descriptions, HiL mappings, and symbolic grounding of conversations. Exact prompts used for self-verification are in appendix 4.15.

Multi-Agent Verification We further improve our refinement process via a principled and highly interpretable multi-agent verification framework. AgentEval ([111]) is a multi-agent framework for evaluating task utility. It takes the task description and 1-2 examples of successful and unsuccessful task runs as input. It generates a categorical rubric of criteria that helps determine the success/failure of future task runs.

The framework uses critic, quantifier, and verifier agents. The critic agent generates several criteria to evaluate the examples, the quantifier agent rates the examples on the criteria, and the verifier agent checks the robustness of each generated criterion. We add

an evaluator agent on top of the criteria judgments to decide whether a data sample should be retained or not. We show the workflow charts in Fig. 4.3. We use instances of GPT-4o-mini for each agent in the framework. We provide the criteria generated for each evaluation workflow in appendix 4.17.

4.3.3 Comparison with Existing Norm Datasets

Three of the prominent existing works that address social norms in conversation are NORMSAGE ([18]), NORMDIAL ([17]) and NORMBANK ([19]).

NORMSAGE presents a framework to obtain norms using LLMs and then perform self-verification. Their main contribution is the pipeline rather than a dataset. In contrast, we gather norm descriptions, operationalize a structured organization and filtering pipeline that efficiently uses and amplifies culturally proficient human annotation.

NORMDIAL is closest to our work as they propose a human-in-the-loop approach with LLM to obtain situation-specific norms. However, we introduce *norm concepts* that aggregate situation-specific descriptions into more general behavioral concepts. We also present robust automated verification, which is evaluated using multi-point human evaluation. The scale of our dataset is larger.

NORMBANK uses situational attributes that align with the factual segment of our schema. However, unlike our work, these norms are not derived from real conversations but are instead defined by a list of high-level scenarios. Our dataset focuses on capturing nuanced human behavior within a specific culture, whereas NORMBANK aims for broader, more general cultural coverage.

4.4 Qualitative Evaluation

Our cultural context dataset is prone to reliability issues from many sources, such as LLM hallucinations and scaling errors in the HiL process. Hence, we employ automated verification to rectify these errors. Now, we evaluate the quality of the obtained data and the refinement techniques via human evaluation. This allows us to quantify the reliability of the cultural context dataset as a resource. To that end, we obtain human judgments for several aspects of

	Relevance		Mapping		Violations	
	qual	**ret**	**qual**	**ret**	**qual**	**ret**
LLMs	81	-	91	-	60.3	-
Self-Ver	82.2	73	93.4	85.6	64.3	74
MultiAgent	**88.4**	**91.3**	**94.8**	**93.8**	**66.1**	**81.4**

Table 4.3. Summary of human evaluation results of pipeline and refinement techniques. qual(ity) - % of correct samples in the refined data; ret(ention) - % of correct samples which passed refinement

the dataset on a sampled subset. We observe that the automated verification-based filtering strategies significantly improve the dataset's quality. AgentEval's criteria-based refinement outperforms the self-verification strategy in all aspects.

Ideally, refinement techniques should retain *good* samples while discarding *bad* samples. Hence, we focus on two metrics for our evaluation: *quality* and *retention*. *Quality*, which is analogous to precision, is defined as the percentage of *good* samples in the post-refinement data. *Retention*, analogous to recall, is defined as the percentage of original *good* samples retained after refinement.

We randomly sample 726 description across 239 conversations for human evaluation annotation. Out of 726 examples, 580 are mapped to created norm concept structures. We focus our evaluation on three aspects: LLM hallucinations, symbolic annotation, and HiL scaling errors. Hence, we ask humans to (1) judge the *relevance* of the LLM-generated description to the conversation, (2) judge whether the mapping of the norm concept to the description is accurate, and (3) judge whether or not the norm concept is violated in that particular conversation. Each of the three decisions is a binary yes/no answer. We use three native culture annotators for these tasks. All annotators are graduate students who are CSS researchers as well.

We evaluate the initial LLM generations from §4.3.2, self-verification filtered, and AgentEval filtered datasets against human judgment data. We present the results in Tab. 4.3. We observe that the quality of the dataset improves upon self-verification in all three aspects. It further improves significantly upon multi-agent verification. We reduce hallucinations in

norm description generations from 19% to 11.6%, improve description-norm concept mapping quality from 91% to 94.8%, and improve violation status quality from 60.3% to 66.1% using multi-agent verification refinement. We also note that multi-agent verification retains a significantly higher portion of data than self-verification. For norm descriptions, we retain 91.3% of correct descriptions (vs. 73% by self-verification); for norm concept mapping, we retain 93.8% (vs. 85.6%); and for violation judgments, we retain 81.4% (vs. 74%) data.

IAA: We also measure inter-annotator agreement to quantify the subjectivity of these tasks. For concept mapping, we obtain Krippendroff's alpha of 0.61, which points to moderate to strong agreement. For hallucinations, we obtain 0.74, and for violation status, we obtain 0.68. We also ask annotators to rate k-NN augmentations on how relevant they are to the assigned norm concept on a Likert scale of 1-5. This resulted in an average Likert score of 4.11. These results demonstrate high agreement and a high success rate of k-NN augmentation. We further present interesting qualitative visualizations of norm concepts in appendix 4.11.

4.5 Downstream Task Evaluation

Our experiments aim to evaluate the *usefulness* of cultural information and graph-based schema. We use empirical performance on conversational understanding tasks as a proxy for *usefulness*. We focus on 2 classes of models: no-context models and cultural context models. Cultural context models are further divided into 2 types: models that consume cultural context as *text* and as *a graph*. We use 3 models for our experiments: RoBERTa ([120]), LLama-3.1 ([114]), and our ConvGraph model. We design a graph model that uses PLM as a node encoder and leverages the schema structure in Fig.4.1(a). We briefly discuss the datasets and tasks and then models used.

Datasets: We perform experiments on 3 tasks across 3 existing datasets. We conduct experiments with MPDD, CPED, and LDC CCU Chinese datasets (§4.2). We use emotion detection (all 3 datasets), sentiment detection (CPED), and dialogue act identification (LDC CCU, CPED) tasks to benchmark the models.

Tasks: For emotion detection, MPDD is annotated using 7 emotion labels, CPED with 13, and LDC CCU ZH with 9 labels. Dialogue act identification in CPED is a 19 class task, and the LDC CCU Chinese dataset uses 10 labels. CPED sentiment is a 3-way annotation. We use the original train/validation/test splits for CPED and MPDD datasets. For the LDC CCU Chinese dataset, we use the LDC2022E18 release as the train set, the LDC2023E01 release as the validation set, and the LDC2023E20 release as the test set. In this dataset, *neutral* label for emotion detection and *other* for dialogue act identification are over-represented. We also find that several samples from these classes are missed annotations. Hence, to avoid skewing the models, we down-sample these classes in all data splits to 1% of the actual labels. This makes these classes roughly the same size as the other classes.

4.5.1 Models

We experiment with 3 models: RoBERTa, LLama-3.1-8B-Instruct, & our ConvGraph model.

RoBERTa: We fine-tune the RoBERTa-Chinese-WWM-base ([120]) model on each task as sequence classification. We provide all the previous turns in the conversation and the current turn as inputs and predict class labels for respective tasks. We use cross-entropy loss to train the model. We use loss weighting to deal with class imbalance. Further details are provided in appendix 4.12. We use the HuggingFace transformers library ([84]) for all our experiments.

LlaMa-3.1-8B-Instruct: We perform QLoRA ([115]) fine-tuning of state-of-the-art LLM, Llama-3.1-8B-Instruct model using the same inputs as RoBERTa-Chinese-WWM model for the no-context experiments. Then, for cultural context experiments, we augment both factual component information and cultural information as text. We provide the exact input format in appendix 4.14.

ConvGraph Model: For the no-context model, we create a graph from only conversational turns. The graph consists of one node with the conversation text. This node is connected to several child nodes with one turn each. We perform tasks as node classification. We use RoBERTa-Chinese-WWM-base to encode the dialogue and turn nodes. We use DGL ([121])

library to implement the graph model. We use two GraphSAGE ([122]) layers on top of the PLM encoders and then pass the node embedding to a final GraphSAGE layer for final task classification.

For the ConvGraph + cultural context model, we use the graph structure presented in Fig. 4.1(a). To encode cultural context nodes, which are in English, we use RoBERTa-base ([123]) as node encoder. We represent all edges as bi-directional edges. In this model, the contextual nodes such as norm concepts, relationships, settings, etc, are shared across conversations. This makes the entire data split (train/valid/test) into a single graph. We use randomly sample a neighborhood of 10 for each node during training and inference to make the computation tractable.

4.6 Results

Task/ Model	MPDD	CPED			LDC CCU	
	Em	Em	Sent	DA	Em	DA
	w-F1	w-F1	w-F1	w-F1	w-F1	w-F1
No Context Models						
RoBERTa	61.13	20.89	46.69	55.40	56.50	64.84
ConvGraph	61.81	21.42	47.66	56.28	54.83	65.62
Llama-3.1-8B-Instruct	45.48	24.73	53.91	54.15	60.11	67.15
Cultural Context Models						
Llama-3.1-8B-Instruct + Context	48.40	**29.81**	**55.00**	**60.16**	**62.16**	68.28
ConvGraph + Context	**64.34**	21.65	49.90	57.24	57.97	**71.74**

Table 4.4. Results on test sets. We report weighted F1 scores. Em-emotion, Sent-sentiment, DA-Dialogue Act.

We present the results on the test sets of each task of our experiments in Tab.4.4. We observe that the cultural context models outperform no-context models significantly in all the tasks. We report the frequency weighted-F1 score metric.

We mainly note the performance of cultural context models. Both Llama-3.1-8B and ConvGraph models perform significantly better when the cultural context information is augmented. It is interesting to note that despite significant pre-training and post-training on extensive data, llama-3.1 models still benefit significantly from cultural context information.

For the llama-3.1-8B model, performance on CPED emotion improves from 24.73 to 29.81, and CPED dialogue act performance improves from 54.15 to 60.16 F1.

It is also interesting to observe that the ConvGraph + cultural context model performs best on two tasks: MPDD emotions and LDC CCU dialogue acts. ConvGraph model only has $\approx$ 500M parameters as opposed to Llama-3.1-8B.

It is intriguing to note that the Llama-3.1-8B model significantly lags behind even the RoBERTa baseline on the MPDD emotion task. We posited that this might be due to a lack of robust multi-lingual capabilities. Hence, we trained the Llama-3.1-70b model on this task as well. But, it also reaches only 53.41 F1 on the task. We conclude that this might be due to the nature of the Llama class of models pre-training, post-training, and the annotation distribution for this particular dataset.

4.7 Conclusion

We propose a novel cultural context schema grounding pipeline. We introduce *Norm Concepts* which abstract over cultural beliefs. Using the pipeline and LLMs, we create high-quality cultural context data and perform human evaluation. Finally, we show that the dataset improves conversational understanding empirically.

4.8 Acknowledgments

We thank the reviewers for their insightful comments that helped improve the paper. This work was supported by NSF CAREER award IIS-2048001 and the DARPA CCU program. The contents are those of the author(s) and do not necessarily represent the official views of, nor an endorsement by, DARPA, or the US Government.

4.9 Limitations

We present a novel framework built upon LLMs and a human-in-the-loop approach. Both these approaches are prone to bias due to data and human components. Our evaluation is qualitative and hence relies on heuristic metrics. Our method requires expensive data collection and annotation protocol. We perform annotations for only one language. Cultural

norm discovery using LLMs is limited due to the lack of depth on knowledge on some cultures for LLMs. Western bias due to training data might propagate into the dataset. This is a pioneering attempt to build large-scale datasets and hence requires careful usage of the data and technology. Our verification strategy also constitutes LLMs and hence that data still contains close to 10% noisy data event after validation.

LLM-generated cultural norms could potentially perpetuate stereotypes, contain hallucinations, bias, or other harmful distributional characteristics that could be challenging to detect and filter. In this work, we address these issues such as hallucinations, and stereotypes by (1) involving culturally proficient humans in the norm concept creation stage and (2) also performing principled automated filtering which is again evaluated against human annotation. In our pipeline, humans are not just aggregating the descriptions that they see, but rather they are using their cultural expertise to identify norm concepts that are prominent in their culture and are also supported by data. Despite these steps, the output is not necessarily completely absolved of these issues. On the other hand, human-generated datasets have also been shown to contain harmful biases ([124–129]). LLMs are powerful NLP tools that allow us to explore tasks that were not practical before their advent. A case in point is our paper. It would have been highly impractical or hugely expensive to collect culturally proficient human annotations for cultural norms at this scale without LLMs. We believe that a pragmatic approach to this issue would be to carefully document and evaluate biases at each stage of the pipeline and invest community effort in building de-biasing techniques, bias-free training approaches, and so on. This could potentially lead us to creating fair(er) systems. We believe that a pragmatic approach of learning to build better guard rails, mitigating the harmful effects, and being aware of these biases could be a more fruitful path forward.

4.10 Ethics Statement

Cultural norm discovery using LLMs is limited due to the lack of depth on knowledge on some cultures for LLMs. Western bias due to training data might propagate into the dataset. This is a pioneering attempt to build large-scale datasets and hence requires careful usage

of the data and technology. Human annotation using cultural experts is a high-variance process as the experience of culture varies from region to region and person to person. We try to capture a high-level idea of culture in our work. It is possible that the data propagates biases in the society and hence requires careful usage. Our work is aimed to be a research artifact to help foster better models for cross-cultural interaction. This is by no means an end product to be used in large-scale applications.

4.11 Norm Concept Visualization

Further, we present a qualitative visualization of norm concept distribution over conversational settings in Fig.4.4. We present the visualizations for 1) Norm of Formal Address, 2) Norm of Children obeying Parents' wishes and 3) Respecting the Doctor's expertise. We observe that norms are prominent in different fields. Formal address norm is highly prominent in a company setting, while children obeying parents' wishes is more prevalent in family settings. This is in accordance with the general expectations. This shows that the norm concepts capture interesting aspects of conversational context. Norm (3) is highly prominent in family and hospital settings.

It is interesting to note that we only have field values at a conversational level, and hence, there could be multiple fields with the same conversation. The conversations are not segmented neatly on a scene-to-scene basis, especially in the LDC CCU dataset. This explains the noise distribution of certain norms. It is also noteworthy that the norm descriptions are generated based on conversational content. The field of the conversation alone doesn't always determine the conversational content.

4.12 Reproducibility

For our downstream task experiments, we use a community cluster with several 80G and 40G A100 GPUs. We use a learning rate of 1e-4 for llama QLoRa training. We use r=16 with 8-bit quantization. For the graph model, we use the DGL library with distributed training using several A100 GPUs. Our llama training took 12 hours on average for each task. We ran several development iterations before arriving at the final prompt structure.

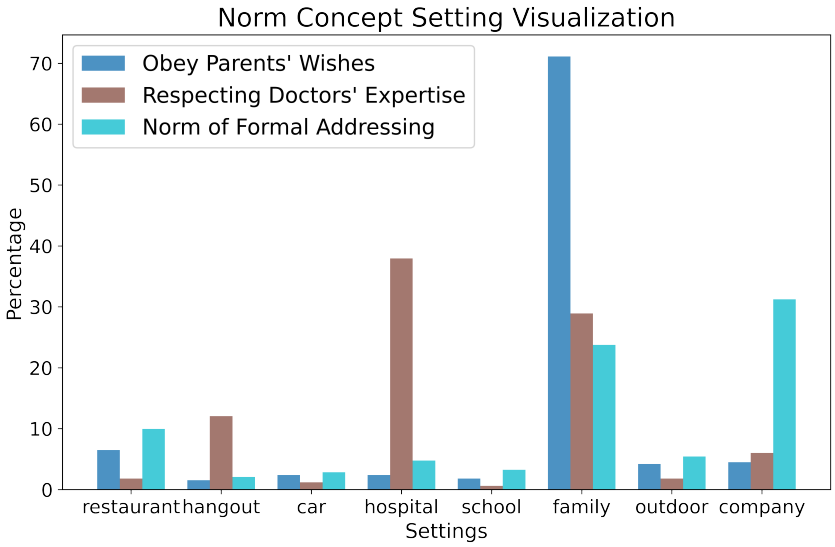


Figure 4.4. Comparison of Conversation Field Distribution Across Various Norm Concepts

The graph model training took 4 hours on average for each task, when augmented with cultural schema information. RoBERTa models take 10-30 minutes to converge in training depending on the size of the dataset. We will release all our code and datasets under MIT license upon acceptance. We extensively use ChatGPT, Claude, Github co-pilot for our coding requirements. We paid ∼400 USD for using OpenAI API for data collection and experiments. We use Grammarly to aid in draft improvement.

4.13 Annotation Guidelines

All annotators used for both HiL concept discovery and human evaluation are graduate students, 3 of whom are also CSS researchers. We have a quid-pro-quo data annotation system in place in our lab. We contribute hours of annotations to other projects and we can get similar hours back when we need them. We use 3 native Chinese annotators, who were born and raised in mainland China for our annotations. We also use 2 non-Chinese annotators. They help in book-keeping tasks in the norm concept generation. They moderate and organize the interactive process sessions which are collaborative. The HiL process described in

§4.3.2 is a collaborative annotation process. All the annotators discuss and reach a consensus before adding a norm concept to the database. In contrast, human evaluation annotation described in §4.4 is an individual annotation task. Hence, we report inter-annotator agreement for these judgments. This annotation is performed by native culture annotators.

4.14 Downstream Task Setup

LlaMa-3.1-8B-Instruct + Cultural Context Model Input:

Instruction: You are a helpful assistant who predicts the <task> of the conversational turn in a Chinese conversation. You are given the cultural context surrounding the conversation, the prior conversation, and the current turn. Predict the <task-name> of the current turn. Choose one of <label set>.

Prior Conversation:

<turn 1>

<turn 2>

Conversation Settings:

Synopsis: ..

Speaker Count: ..

Speaker Sex: ..

...

Relevant Cultural Information:

1) **Norm Concept**:

Theme: <theme-name>

Description: ...

Settings: ..

Violation Sketch: ...

Specific Norm: <norm-description>

2) **Norm Concept**:

...

Potential Violations:

1) <violation-description-1>

..

Current Turn:

<turn-n>

<task-name>:

Instruction and final <task-name> prompt are not provided for the RoBERTa model.

LlaMa-3.1-8B-Instruct, RoBERTa Models Input:

Instruction: You are a helpful assistant who predicts the <task-name> of the conversational turn in a Chinese conversation. You are given the cultural context surrounding the conversation, the prior conversation, and the current turn. Predict the <task-name> of the current turn. Choose one of <label set>.

Prior Conversation:

<turn 1>

<turn 2>

Current Turn:

<turn-n>

<task-name>:

4.15 LLM Prompts and Generations

Prompts used to extract Conversation-Specific Social Norms and Violations from GPT-3.5

1. Translate this conversation into English.
2. List the people mentioned in the conversation and the social relationships between them.
3. List the Chinese cultural norms applicable to this situation. Are there any cultural norm violations observed in this situation? If yes, list them. List the observed and potential effects by index for each violation.
4. Summarize the conversation in 3-4 sentences.

Input

Translated Conversation:

Mrs. Zuo: What is that foolish girl worth giving anything to? Zho Zpeng is such a person!

Mr. Zuo: Oh, wife, why do you always say things that are not conducive to unity? It's reasonable for him to go and give something to his classmate!

Zho Zpeng: Dad, Mom, I'm back!

Mrs. Zuo: I'm afraid you've been drugged and your soul has been seduced by that foolish classmate of yours!

Mr. Zuo: Wife, do you sound like a mother when you speak like that? Why do you always oppose their relationship?

Zho Zpeng: Dad, Mom, please stop talking. I have my own opinions on this matter. I know what I should do and what I shouldn't do. I will never act recklessly. Please trust me!

LLM Description Generations

Summary:

The conversation revolves around Zho Zpeng, his parents, and his relationship with a classmate. Mrs. Zuo disapproves of the classmate and questions the worth of giving her anything, while Mr. Zuo defends his son's actions. Zho Zpeng expresses his independence and asserts that he knows what he should and shouldn't do, asking his parents to trust him.

Norms:

Respect for parents: Filial piety and respect for parents are highly valued in Chinese culture. Children are expected to listen to and obey their parents' opinions and decisions.

Unity within the family: Maintaining harmony and unity within the family is important in Chinese culture. Conflicts or disagreements that disrupt family unity are generally discouraged.

Social relationships and obligations: Chinese culture places importance on maintaining positive relationships and fulfilling social obligations, such as giving gifts to classmates or friends.

Violations:

Disrespectful language: Mrs. Zuo uses derogatory language to refer to Zho Zpeng's classmate, which shows a lack of respect for others.

Opposition towards Zho Zpeng's relationship: Mrs. Zuo continuously opposes the relationship between Zho Zpeng and his classmate, going against the cultural norm of respecting children's autonomy and choices in relationships.

Effects:

Disrespectful language - Observed effect: It can create tension and animosity between Mrs. Zuo and Zho Zpeng.

Opposition towards Zho Zpeng's relationship - Observed effect: It causes disagreement and arguments between Mrs. Zuo and Mr. Zuo, highlighting a lack of unity within the family.

4.16 Symbolic Annotation Analysis

Symbolic Annotation & Verification Prompts

As a two-step cultural and social norms analysis assistant, your task is to evaluate a provided conversation from the context of Chinese culture based on a given social norm and a corresponding norm concept. Your analysis should be comprehensive, considering factors such as age, relationships, settings (e.g., work, family, friends), and the topic of the conversation.

Steps for Analysis:

1. **Evaluate Social Norm and Norm Concept Compatibility** - **Task:** Judge whether the provided social norm aligns with the given norm concept. - **Action:** State if the social norm **matches** or **doesn't match** the norm concept. - **Justification:** Provide a concise reason for your judgment.
2. **Assess Relevance to the Conversation** - **Task:** ..
3. **Determine Social Norm Violation** - **Task:** Judge whether the social norm was **adhered to** or **violated** in the conversation. - **Justification:** Provide a concise reason for your judgment.
4. **Annotate Conversation-Specific Details**

- **Enactor Role:** ...
- **Acceptor Role:** ...

5. **Violation Analysis** *(Only if a violation occurred)* If the norm was violated, provide the following additional details: - **Violating Action:** A brief description of the action that caused the violation (e.g., "badmouthing parents").

- **Violator Role:** ..
- **Victim Role:** ..
- **Violator Emotion:** ..
- **Victim Emotion:** ..

Response Format: Your response must adhere to the format below:

Social Norm - Norm Concept Compatibility: <match/doesn't match> Compatibility Justification: <short justification>

Only if compatible Relevance: <relevant/irrelevant> Relevance Justification: <short justification>

Enactor Role: <sitatuion specific social role, strictly not the name, of the person> Acceptor Role: <sitatuion specific social role, strictly not the name, of the person>

Violation Status: <adhere/violate> Violation Status Justification: <short justification>

Only if a violation occurs

Violating Action: <short phrase>

Violator Role: <sitatuion specific social role, strictly not the name, of the person>

Victim Role: <sitatuion specific social role, strictly not the name, of the person>

Violator Emotion: <one of 9 basic emotions>

Victim Emotion: <one of 9 basic emotions>

Input:

Conversation from Chinese culture:
Zuo Zhengpeng: Lihua, how much longer until the baby is due?
Xu Lihua: Probably next week!
Zuo Zhengpeng: Oh, the baby is moving!
Xu Lihua: Look at you, how embarrassing!
Xu Lihua: Whats the rush? We still have a few days. Well talk about it then. Youre eager to become a father, arent you?
Zuo Zhengpeng: No! By then, it might be too late. Its better to go to the hospital a few days early. Transportation here is inconvenient. If anything happens, we could be in big trouble.

Social Norm:
Respect for elders: It is common in Chinese culture to show respect to older family members, such as parents and grandparents.

Norm Concept Name: Respect for family elders
Norm Concept Description: Respecting the wisdom, experience, and authority of elder members in the family hierarchy.
Norm Concept Potential Violation Sketch: Showing disrespect and ignoring suggestions and advice of any elder members of the family.
Norm Concept Scenario: family
Enactor Role: any younger family member
Acceptor Role: elder family members such as parents, uncle, grandparents, etc.

GPT Output:

Norm-Concept Compatibility: match

Relevance: relevant
Relevance Justification: The conversation takes place within a family context, discussing the impending birth of a child, which inherently involves respect for family roles and responsibilities.

Enactor Role: younger family member
Acceptor Role: elder family member

Quality Judgment: accurate
Justification: The annotations correctly identify the compatibility of the social norm with the norm concept, the relevance to the family context, and the adherence to the norm in the conversation, reflecting a respectful interaction between the younger and elder family members.

4.17 AgentEval Task Criteria

4.18 Data Sources and Statistics

4.19 Annotation GUI

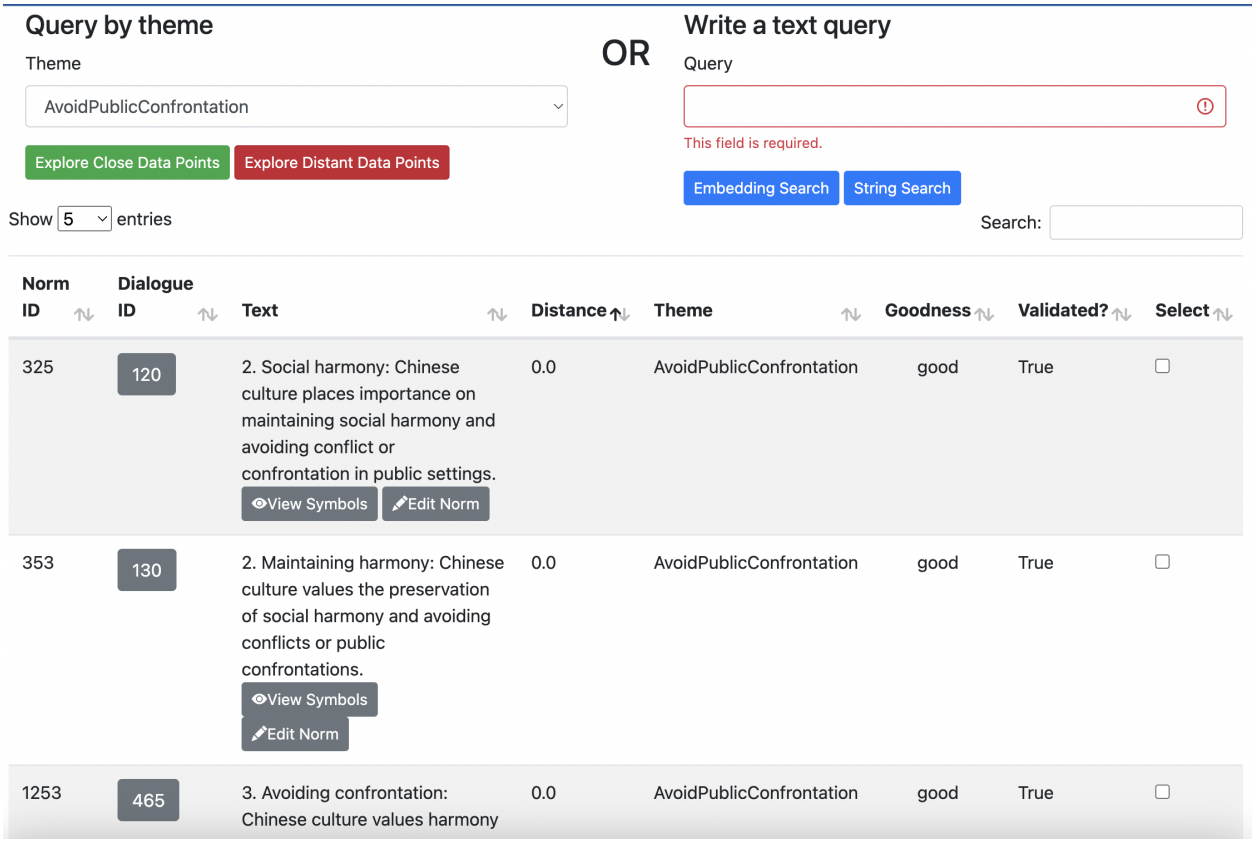


Figure 4.5. Annotation Interface for Norm Concept Discovery

4.20 Norm Concept Visualization

4.21 Schema Example

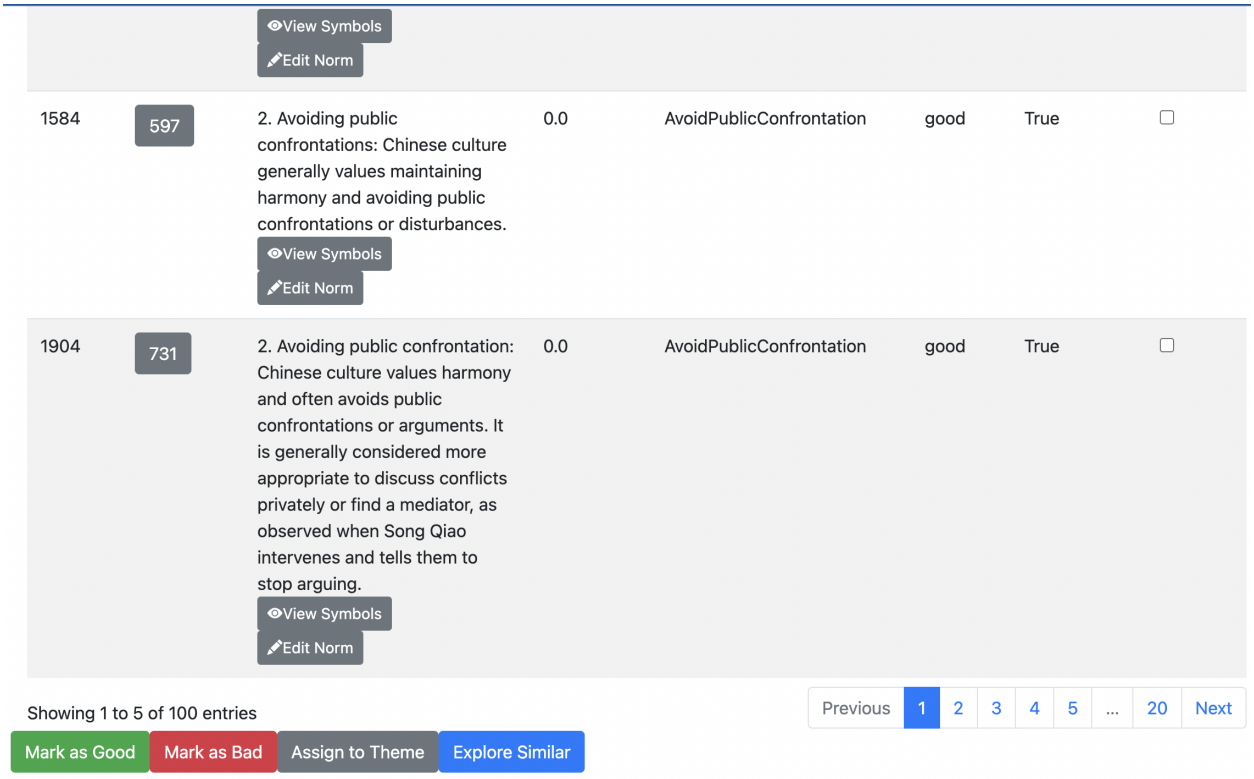


Figure 4.6. Annotation Interface for Norm Concept Discovery

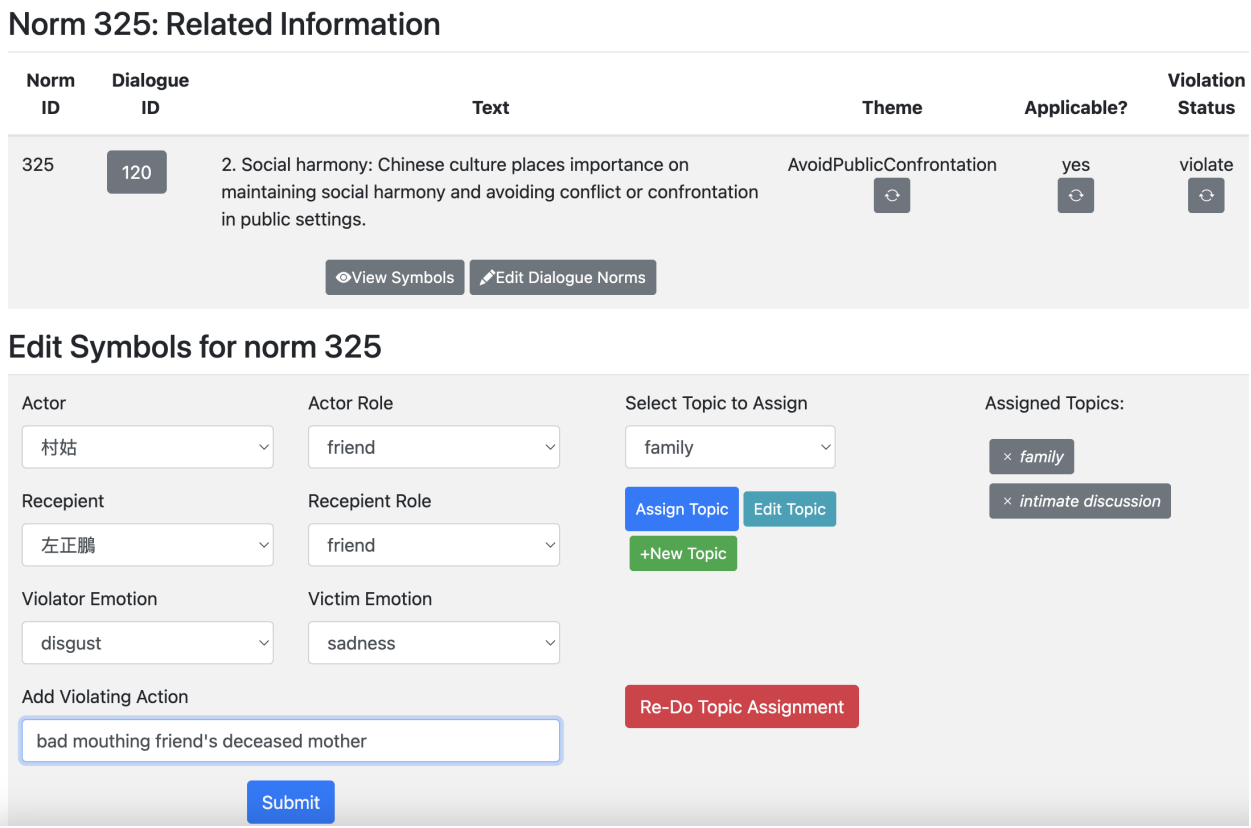


Figure 4.7. Human Validation and Symbolic Annotation of Cultural Context Annotation Framework

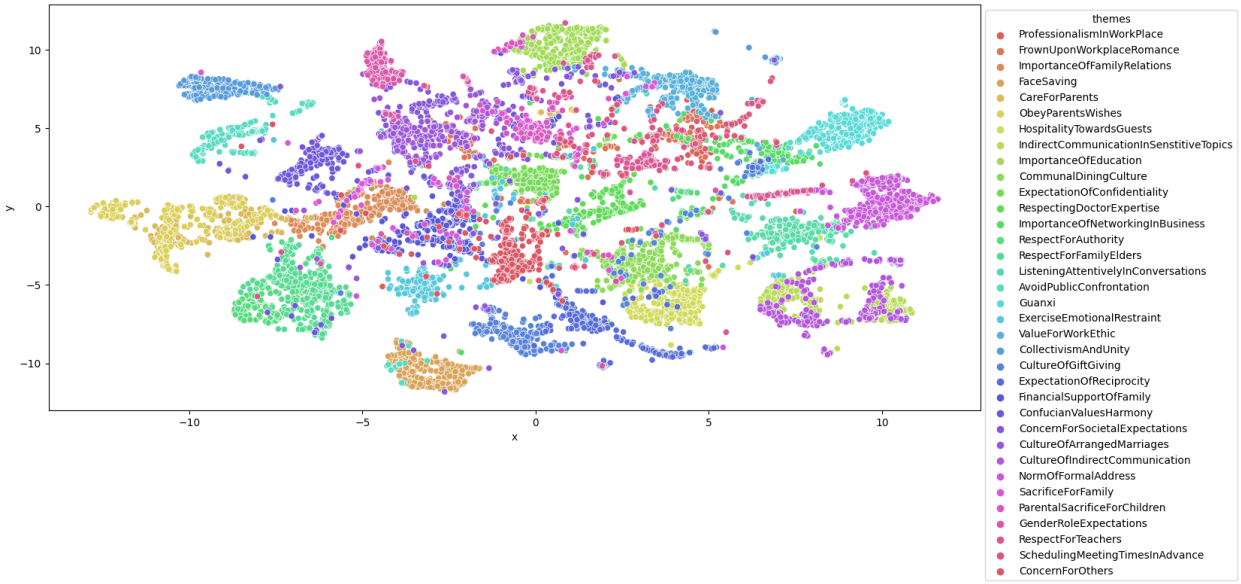


Figure 4.8. Norm Concept Visualization

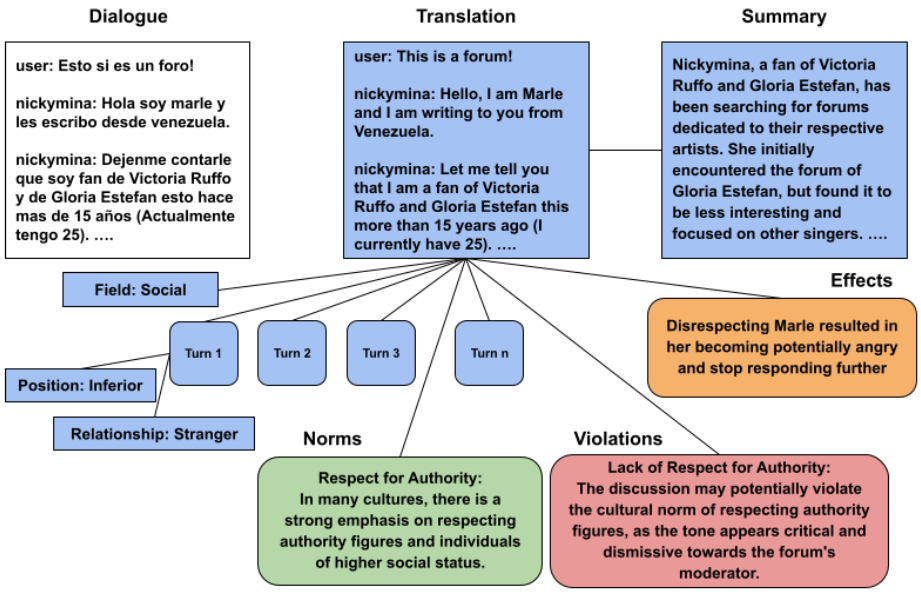


Figure 4.9. An Example Instance of Schema Augmented Conversation

Criteria	Description	Accepted Values
Task: Relevance Judgment		
Clarity	The degree to which the reasoning behind the relevance judgment is clear and understandable.	1 - very unclear, 2 - unclear, 3 - neutral, 4 - clear, 5 - very clear
Contextuality	The extent to which the judgment considers the specific context of the conversation including relationships, settings, and cultural nuances.	1 - not contextualized, 2 - poorly contextualized, 3 - moderately contextualized, 4 - well contextualized, 5 - very well contextualized
Appropriateness	How well the social norm applies to the situation discussed in the conversation.	1 - not appropriate, 2 - minimally appropriate, 3 - somewhat appropriate, 4 - appropriate, 5 - highly appropriate
Cultural Relevance	The degree to which the social norm reflects significant aspects of Chinese culture relevant to the conversation.	1 - not culturally relevant, 2 - minimally relevant, 3 - somewhat relevant, 4 - culturally relevant, 5 - highly culturally relevant
Consistency	The consistency of the judgment with established norms and expectations in Chinese cultural contexts.	1 - highly inconsistent, 2 - inconsistent, 3 - neutral, 4 - consistent, 5 - highly consistent
Relationship Dynamics	The consideration of the relationships between the individuals involved in the conversation and how that influences the relevance of the social norm.	1 - not considered, 2 - weakly considered, 3 - moderately considered, 4 - strongly considered, 5 - very strongly considered
Setting Analysis	Evaluation of how the setting of the conversation (e.g., family, workplace, casual gathering) impacts the relevance of the social norm.	1 - no impact, 2 - minor impact, 3 - moderate impact, 4 - significant impact, 5 - critical impact
Norm Visibility	The extent to which the social norm is visible or recognized in the context of the conversation.	1 - not visible, 2 - slightly visible, 3 - moderately visible, 4 - visible, 5 - very visible
Social Hierarchy	How well the judgment incorporates aspects of social hierarchy or status within the Chinese cultural context.	1 - not addressed, 2 - poorly addressed, 3 - addressed, 4 - well addressed, 5 - very well addressed
Timeliness	Consider how the relevance of the social norm may change over time or in different historical contexts.	1 - not timely, 2 - slightly timely, 3 - moderately timely, 4 - timely, 5 - very timely
Evidence	The amount and quality of evidence or examples provided to support the relevance judgment.	1 - no evidence, 2 - minimal evidence, 3 - some evidence, 4 - good evidence, 5 - strong evidence
Norm Specificity	The degree to which the social norm discussed is specific and detailed rather than vague.	1 - very vague, 2 - vague, 3 - somewhat specific, 4 - specific, 5 - very specific
Emotional Tone	Evaluation of how the emotional tone of the conversation affects the relevance of the social norm.	1 - negative impact, 2 - slight negative impact, 3 - neutral impact, 4 - slight positive impact, 5 - positive impact
Discourse Style	The impact of the discourse style (formal, informal, persuasive, etc.) on the relevance of the social norm.	1 - no impact, 2 - minimal impact, 3 - moderate impact, 4 - significant impact, 5 - critical impact
Dissonance Level	The level of dissonance between the social norm and the expressed opinions or behaviors in the conversation.	1 - highly dissonant, 2 - moderately dissonant, 3 - neutral, 4 - somewhat aligned, 5 - highly aligned
Cultural Evolution	Consideration of how modern trends or changes in society may affect the relevance of traditional social norms.	1 - outdated, 2 - slightly outdated, 3 - somewhat relevant, 4 - relevant, 5 - highly relevant
Collective Perspective	How well the judgment considers the views of the collective society versus individual perspectives.	1 - individual-focused, 2 - slightly collective, 3 - moderately collective, 4 - largely collective, 5 - entirely collective
Norm Adoption	Recognition of how widely the social norm is adopted or practiced within Chinese society.	1 - not adopted, 2 - rarely adopted, 3 - somewhat adopted, 4 - widely adopted, 5 - universally adopted
Age Sensitivity	The extent to which the relevance judgment is sensitive to the age differences of the individuals involved.	1 - not sensitive, 2 - slightly sensitive, 3 - moderately sensitive, 4 - sensitive, 5 - very sensitive
Analytic Depth	The depth of analysis provided in the reasoning for the relevance judgment.	1 - very superficial, 2 - superficial, 3 - moderate depth, 4 - deep, 5 - very deep
Feedback Responsiveness	The degree to which the judgment accounts for feedback or reactions from the individuals in the conversation.	1 - not responsive, 2 - slightly responsive, 3 - moderately responsive, 4 - responsive, 5 - very responsive
Rule Conformity	How closely the judgment conforms to established social rules and expectations within the context.	1 - highly non-conformant, 2 - non-conformant, 3 - somewhat conformant, 4 - conformant, 5 - highly conformant

Table 4.5. Evaluation Criteria Generated by AgentEval for *Relevance* of *Norm Description* to *Conversation* Judgment

Criteria	Description	Accepted Values
Task: Norm Interpretation and Evaluation		
Contextual Influence Evaluation	Evaluate how the context (e.g., setting, relationships) influences the interpretation of the norm.	high influence, medium influence, low influence, no influence
Cultural Appropriateness Assessment	Judge whether the norm is appropriate within the cultural context of the conversation.	appropriate, inappropriate
Generational Perspective Evaluation	Assess how generational differences impact the conversation's norms.	significant difference, some difference, no difference
Emotional Weight Assessment	Determine the emotional significance of the norm within the conversation.	high weight, medium weight, low weight
Implications of Norm Violation	Assess the potential consequences of violating the social norm.	severe consequences, moderate consequences, mild consequences, no consequences
Responsibility Assessment	Evaluate who holds the primary responsibility for adhering to the social norm in the conversation.	enactor, acceptor, both
Social Norm Awareness Evaluation	Judge the awareness of the social norm by the involved parties.	fully aware, somewhat aware, not aware
Feedback Necessity Assessment	Determine if feedback is necessary based on the adherence or violation of the norm.	necessary, not necessary
Coping Strategy Evaluation	Evaluate the coping strategies employed by individuals in response to a norm violation.	effective, somewhat effective, ineffective
Communication Style Analysis	Analyze the communication styles used in relation to the social norm.	formal, informal, assertive, passive, aggressive

Table 4.6. Evaluation Criteria Generated by AgentEval for *Symbolic Annotation Quality* Judgment

Dataset	Turn-Level				Conversation-Level									Corpus	
	Em.	DA	Sent.	Sp.-List. Reln.	Summ.	Norms			Violations			Effects		Field	Norm Concepts	
						Desc	Valid	Symb. Attr	Desc	Valid	Symb. Attr	Desc	Valid		Conc.	Assg. Norms
MPDD	A	LM	-	A	GPT	10,637 (GPT)	900 (H)	726 (H)	5,721 (GPT)	213+65915 (H+GPT)	66,128 (H+GPT)	14,521 (GPT)	213 (H)	4,141 (A)	35 (H)	422 (H)
CPED	A	A	A	LM	GPT	32,209 (GPT)			17,865 (GPT)			25,866 (GPT)		11,832 (A)		
LDC CCU	A	A	-	LM	GPT	23,306 (GPT)			15,605 (GPT)			14,908 (GPT)		7,554 (A, LM)		36,954 (k-NN)

Table 4.7. Sources and Counts of Collected Schema Grounding Dataset. A - gold annotation; LM - Llama-70b generated; GPT - GPT-3.5 generated; H - human annotation; kNN - interactive k-nearest neighbors; Em - emotion; DA - dialogue act; Sent - sentiment; Sp_List Reln - speaker_listener relationship; Desc - descriptions; Symb_Attr - symbolic attributes; Valid - Validated; Conc - concepts; Assg_Norms - norms assigned to concepts;

5. CONTEXTUALIZED LLM

Large Language Models (LLMs) have demonstrated impressive semantic understanding and reasoning capabilities on NLP tasks ([130]). LLMs are subject to extensive pre-training and post-training procedures that result in these capabilities ([131–135]). These training pipelines are typically highly intensive in terms of data and compute.

In addition to parametric knowledge, various NLP tasks also require access to external information that is unavailable at training time. Examples of such information could be news, new information from the web, latest data from social media, etc. Fine-tuning massive LLMs to infuse new information regularly is highly impractical. Hence, to address this challenge, the paradigm of Retrieval Augmented Generation (RAG) models has emerged.

RAGs typically index external information in embedding space. Based on the input prompt (query), they employ search over the indexed information. This information is then filtered, formatted, and augmented into the LLM input to allow the model to perform tasks that require access to external information.

Computational Social Science (CSS) deals with problems relevant to modeling social understanding and human behavior. Several past works observe that using structured social information such as relationships, power dynamics, attitudes, preferences, and social structures, etc, could significantly boost the performance on CSS tasks ([136–140]).

Humans adapt to social settings via a "*mental representation of past events that is integrated with other memories into a general mental structure and guides the processing of future social cues*" ([141]). Such a mental structure could typically be envisioned as an efficient arrangement of rich and highly structured information.

While LLM pre-training and post-training procedures help them obtain useful understanding of social phenomena and human behavior, we argue that this is not entirely sufficient. Several past works observe significant performance gains by explicitly providing structured social information at inference time ([137, 142–144]). Existing RAG pipelines facilitate one way of augmenting such information in a principled manner. But, these pipelines are mainly designed with factual information in mind. In contrast, social context information is utilized differently by the models.

Model	MPDD Emotion	LDC CCU Emotion	LDC CCU Dialogue Act	CPED Emotion	CPED Dialogue Act	CPED Sentiment
Llama-3.1-8B-In	30.86	42.24	48.72	14.70	4.94	45.92
Llama-3.1-8B-In + Ctx	26.53 (-4.33)	40.62 (-1.62)	44.35 (-4.37)	26.03 (+11.33)	11.97 (+7.03)	46.41 (+0.49)
Llama-3.1-8B-In + QLoRA	45.48	60.11	67.15	24.73	54.15	53.91
Llama-3.1-8B-In + QLoRA + Ctx	**48.40 (+2.92)**	**62.16 (+2.05)**	**68.28 (+1.13)**	**29.81 (+5.08)**	**60.16 (+6.01)**	**55.00 (+1.09)**

Table 5.1. Weighted-F1 performance comparison across six tasks for Llama-3.1-8B-Instruct model variations. Best scores in each task are highlighted.

In this paper, we mainly address this research question – **Can we better align knowledge augmentation techniques in LLMs with social context understanding goals?** To that end, we propose a deeply integrated module into the transformer encoder-decoder architecture ([145]), to augment contextual information. We call this a *composer* module. The proposed architecture is shown in Fig.5.1. We describe the architecture in detail in §5.4.

First, we motivate the need for a better contextual understanding model using a quantitative experiment on conversational understanding tasks. We show that while the contextual information is useful, using text based input format doesn't allow the model to fully leverage the contextual information. This leads us to the proposed *context composer* augmentation.

The proposed architecture consists of an addition *context composer* module addition to the transformer encoder-decoder model. *Context composer* module consists graph attention module that computes contextual information embeddings leveraging the rich contextual structure. Then, this information is integrated into the encoder embedding via a cross-attention mechanism. These embeddings are then passed to the decoder module for auto-regressive generation.

Then, we evaluate the proposed architecture on political understanding datasets that require holistic contextual understanding. We show that the proposed module augmented with Flan-T5-large model significantly out performs vanilla Flan-T5-large model that is trained on these tasks.

In the rest of this paper, we first describe the data we use for our experiments (§5.2). We discuss the motivating task experiment in §5.3. Then, we detail our proposed architecture in §5.4. Finally, we present our results (§5.6) and conclude with a short discussion of future work directions.

5.1 Related Work

RAGs: Retrieval Augmented Generation paradigm emerged as a solution for tasks where LLMs require external information that is neither part of their pre-training data, nor part of the information provided in the prompt by the user. They typically consist is three phases: retrieval, augmentation, and generation. Retrieval research could be broadly categorized into indexing, pre-retrieval and post-retrieval improvements. Indexing methods that are closely related to our work are metadata-based indexing and entity-based indexing.

Our work could be categorized into research that improves augmentation of external knowledge into RAGs. There are typically two directions of augmentation research: (1) natural language based augmentation and (2) embedding-based augmentation. Natural language augmentation is significantly more popular because of the wide adoption of decoder-only language models. Our work relies on encoder-decoder style generative models. We choose this architecture because the encoder allows us to embed the contextual information in the same representation space as the language model.

Graphs + LLMs: Several works address the challenging task of integrating graphs into LLMs. These works can be largely categorized into training-free and training-dependent approaches. They can further also be classified as graphs as text, LLMs as graph encoders, and graphs as embeddings approaches.

One notable work that is closest to our work is G-Retriever. While they also work with rich, text-attributed graphs, their approach interviews the graph to generate a natural language description that is plugged into the LLM.

LLMs and CSS: LLMs have significantly influenced CSS in NLP. [143] benchmark a host of CSS tasks. But, their main focus is on traditional CSS tasks. Our focus is on specific genre of CSS tasks that could benefit significantly from rich and nuanced contextual information.

5.2 Data

We build an extensive training and evaluation data benchmark of nuanced social context understanding tasks. We combine existing datasets from [136] and [137] for our experiments.

All the tasks are formatted as triplets: ⟨*prompt*, *text-attributed context graph*, *output*⟩. Our datasets are divided into two categories: political understanding and conversational understanding tasks.

Political Discourse Tasks: target entity & sentiment prediction, and vague text disambiguation.

Conversational Understanding Tasks: emotion, dialogue act, and sentiment detection.

We use the conversational understanding tasks for the motivating quantitative experiment discussed in §5.3. We use the political understanding tasks for our evaluation experiments in §5.5.

5.3 Conversation Task Experiment

In this experiment, our main goal is to examine how well contextual information is used by an LLM. We approach this by choosing an existing dataset on which fine-tuning with contextual information improves the empirical performance significantly. Then, we evaluate if the same LLM can use the contextual information sufficiently well in an inference-only setting.

For this experiment, we choose MPDD, CPED and LDC CCU datasets used by [137]. They show that fine-tuning using social context information significantly improves the performance on all the tasks on Llama-3.1-8B-instruct model.

We use the same prompt structure and evaluate the model with and without cultural context information in an inference-only setting. We present our results in Tab5.1. We observe that while fine-tuning improves the performance on all the tasks, inference-only mode only improves performance on the CPED dataset tasks. Our hypothesis is that, while the contextual information has useful data for MPDD and LDC CCU datasets as well, the model is unable to use the context effectively for these datasets. This motivates the need for better augmentation of contextual information in LLMs for tasks that require nuanced understanding of contextual information.

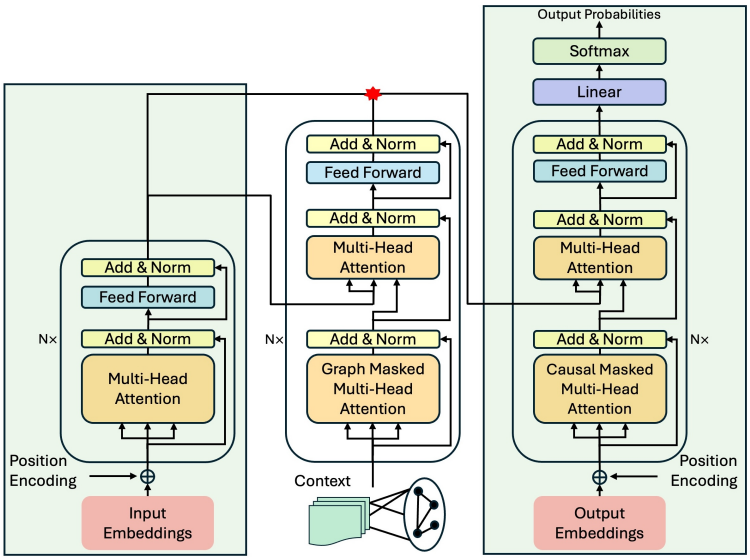


Figure 5.1. Architecture of Proposed Contextualized RAG model

5.4 Proposed Architecture

Our proposed architecture aims to integrate contextual information into the LLM architecture in a deeply integrated manner without adding significant computational penalty. RAGs index documents in the embedding space and then retrieve them using shallow embedding search. Following this, they obtain chunks of relevant contextual information and place it into the prompts, thus leveraging contextual information at inference time [146].

While this paradigm works well when contextual information is limited to a few chunks of highly relevant information, it suffers from several limitations for CSS tasks. As discussed earlier, contextual information in CSS could be vast and potentially noisy. RAGs are known to suffer from lost-in-the-middle problem, sensitivity to noisy contextual information, and replicating contextual information with shallow semantic integration into the generated responses.

To address these issues, we propose a novel *context composer* module addition to the traditional encoder-decoder architecture. The context composer module is an attention-based module that is architecturally analogous to the decoder module. It takes a text-attributed context graph as input and integrates it into the encoder embeddings which are

then passed to the decoder module for further processing. This allows the structure of the contextual information to be represented in the model's processing pipeline.

Architecture Details: The *context composer* module consists of a graph-masked self-attention module followed by an add & norm operation. Then, it consists of a cross-attention module for the context embeddings and encoder embeddings. Then, the output sequence is normalized via residual connection and a layer normalization operation. Following this, the sequence is passed through a position-wise feed-forward layer.

The output sequence from *context composer* module is fused with the output from the encoder module. Hence, there is potential for employing several fusing functions: (1) simple add & norms operation, (2) gated addition operation, (3) parametric gating operation. In the current work, we employ a simple add & norm operation. We leave exploration of fusing operations to future work.

The fused output of the *encoder* and *context composer* modules are then passed to the *decoder* module which generates output probabilities for auto-regressive decoding.

5.5 Experiments

We perform experiments with two datasets released by [136]: (1) tweet target & sentiment detection task and (2) vague text disambiguation task. [136] demonstrate that explicit context modeling task outperform other genre of models including in-context GPT-3 model on their tasks.

We use the baseline results from [136] and perform additional experiments on the GPT-4o model and our proposed architecture instantiated using pre-trained architecture of Flan-T5 model. We use HuggingFace implementation of Flan-T5-large model and implement a Set-T5 model which augments our proposed *context composer* module to the architecture.

We use the training sets of the three tasks to train the model. We freeze the encoder parameters during our training. We train only the *context composer* and the *decoder* modules. We train independently for each of the tasks using their respective training data and we evaluate on the test sets. We use the validation sets for early stopping criteria.

We evaluate against *four* baselines: (1) GPT-3 text context baseline, (2) GPT-4o text context baseline, (3) Discourse Contextualized Framework models from [136], and (4) fine-tuned Flan-T5-large models. The GPT-3 and GPT-4o baselines are training-free approaches but leverage huge and extensively trained LLMs. The DCF baseline is a classifier model, while Flan-T5-large baseline and our augmented model are both generative models comparable with smaller LLMs (SLMs). They consist of close to 800 million parameters.

5.6 Results

Model	Macro-F1 Scores		
	Target Entity	**Target Sentiment**	**Vague Text**
GPT-3 (text context)	69.77	55.00	62.58
GPT-4o (text context)	**76.94**	**71.42**	**75.00**
Contextualized DCF	73.56	65.34	71.71
Flan-T5-large	74.61	67.82	70.98
Ctx Flan-T5 (ours)	76.82	70.51	74.20

Table 5.2. Comparison of Macro-F1 scores across models for Target-Entity identification, Target-Sentiment classification, and Vague Text detection tasks.

We present the results of our experiments in Tab.5.2. We observe that GPT-4o model outperform all the other models on all the tasks. But it is trained on extensive amounts of data and consists of orders of magnitude more parameters than the other baseline models.

Contextualized Flan-T5 model outperforms all models except GPT-4o on all the tasks. Contextualized Flan-T5 model consists of close to 800 million parameters, while GPT-3 consists of 176 billion parameters. Number of parameters on GPT-4o is unknown.

5.7 Future Work

Our work is an initial step towards better context-integrated generative models. While we propose the *context composer* module, we don't fully explore the augmentation of context information with encoder embeddings. Another future work direction could be pre-training the model on large amounts of data to create a general-purpose context-integrated model for

CSS tasks. Large amounts of self-supervised data could be created similar to the learning tasks in [80]. Evaluation of the model on contextual understanding CSS tasks is also an open research problem. While [143] compiles a benchmark for CSS tasks, they don't focus explicitly on tasks that could benefit from rich, nuanced social context information. Hence, such an evaluation benchmark could be quite useful in measuring progress in a standardized manner.

5.8 Conclusion

In this paper, we propose a novel RAG architecture by augmenting existing encoder-decoder architecture of T5 model with an attention-based *context composer* module. We motivate the architecture design via a quantitative experiment on a conversational task dataset. Then, perform experiments on the proposed model using existing political text understanding datasets and show that our model performs almost as well as GPT-4o while containing orders of magnitude smaller number of parameters. We conclude by discussing potential future directions of our work.

6. SUMMARY

In this thesis, we address the problem of contextualized language understanding. First, we motivate the problem using human behavior. Humans use information from various sources such as factual knowledge, commonsense, physical cues, and cultural awareness to comprehend the full meaning of the text in its context.

We summarize existing models and discuss their limitations in addressing the task of contextualizing using a large and noisy contextualizing corpus. Then, we introduce three big ideas and apply them to two domains: (1) Obtaining and organizing relevant contextual information, (2) Proposing and training neural frameworks for the representation learning of text and associated context jointly and (3) Evaluating proposed frameworks on tasks that require holistic contextual understanding capabilities. We focus on US politics and Chinese cultural conversations.

6.1 Organizing Context

In the US political domain, we collect large amounts of data from various sources and organize it into a large heterogeneous graph structure. We propose a querying mechanism to dynamically obtain the requested context.

For the Chinese conversational domain, we show that the contextual information is not explicitly stated in regular discourse. Hence, we leverage LLM generation + human in the loop refinement process to obtain rich contextual information for each conversation. We then propose a schema structure to augment contextual information to the conversation representation.

6.2 Frameworks for Contextual Modeling

We propose a Discourse Contextualization Framework (DCF) which builds upon pre-trained language model embeddings to encode the contextual graphs. We design two unsupervised learning tasks to leverage large amounts of structured contextual data and train

the framework. We show that these representations significantly outperform other models. We also show that they capture nuanced contextual information.

For Chinese conversations, we leverage schema representations for downstream tasks via node classification. We use the rich schema data to improve the performance on these tasks significantly across datasets.

6.3 Evaluation of Social Grounding

For the US political domain, we operationalize two tasks that test the holistic contextual understanding capabilities of NLP models. We benchmark state-of-the-art NLP models and human performance on these datasets. We observe that explicit context models outperform all baselines including LLMs such as GPT-3. But, they still lag behind human performance significantly.

6.4 Future Work

We conclude the thesis by presenting our proposal to integrate DCF into existing LLM architecture without losing knowledge embedded into them via the extensive pre-training process. We demonstrate through some initial results that joint learning helps enhance the performance of the integrated model on social context grounding tasks.

REFERENCES

[1] A. Sennet, "Ambiguity," in *The Stanford Encyclopedia of Philosophy*, E. N. Zalta and U. Nodelman, Eds., Summer 2023, Metaphysics Research Lab, Stanford University, 2023.

[2] P. Bhattacharyya, "Natural language processing: A perspective from computation in presence of ambiguity, resource constraint and multilinguality," *CSI journal of computing*, vol. 1, no. 2, pp. 1–13, 2012.

[3] R. Speer, J. Chin, and C. Havasi, "Conceptnet 5.5: An open multilingual graph of general knowledge," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 31, 2017.

[4] D. Lenat and R. V. Guha, "Cyc: A midterm report," *AI magazine*, vol. 11, no. 3, pp. 32–32, 1990.

[5] C. F. Baker, C. J. Fillmore, and J. B. Lowe, "The Berkeley FrameNet project," in *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, Montreal, Quebec, Canada: Association for Computational Linguistics, Aug. 1998, pp. 86–90. DOI: 10.3115/980845.980860. [Online]. Available: https://aclanthology.org/P98-1013.

[6] F. Hayes-Roth, "Rule-based systems," *Commun. ACM*, vol. 28, pp. 921–932, 1985. [Online]. Available: %5Curl%7Bhttps://api.semanticscholar.org/CorpusID:215992686%7D.

[7] K. Johnson and D. Goldwasser, "Modeling behavioral aspects of social media discourse for moral classification," in *Proceedings of the Third Workshop on Natural Language Processing and Computational Social Science*, S. Volkova, D. Jurgens, D. Hovy, D. Bamman, and O. Tsur, Eds., Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 100–109. DOI: 10.18653/v1/W19-2112. [Online]. Available: https://aclanthology.org/W19-2112.

[8] D. Khurana, A. Koli, K. Khatter, and S. Singh, "Natural language processing: State of the art, current trends and challenges," *Multimedia tools and applications*, vol. 82, no. 3, pp. 3713–3744, 2023.

[9] J. Burgess and A. Bruns, "Easy data, hard data: The politics and pragmatics of twitter research after the computational turn," *Compromised Data. From social Media to Big Data*, pp. 93–111, 2015.

[10] B. P. Zeigler and P. E. Hammonds, *Modeling and simulation-based data engineering: introducing pragmatics into ontologies for net-centric information exchange*. Elsevier, 2007.

[11] H. Chen, C. Yang, X. Zhang, Z. Liu, M. Sun, and J. Jin, "From symbols to embeddings: A tale of two representations in computational social science," *Journal of Social Computing*, vol. 2, no. 2, pp. 103–156, 2021.

[12] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[13] C. Raffel *et al.*, *Exploring the limits of transfer learning with a unified text-to-text transformer*, 2023. arXiv: 1910.10683 [cs.LG].

[14] T. B. Brown *et al.*, *Language models are few-shot learners*, 2020. DOI: 10.48550/ARXIV.2005.14165. [Online]. Available: https://arxiv.org/abs/2005.14165.

[15] H. Touvron *et al.*, *Llama 2: Open foundation and fine-tuned chat models*, 2023. arXiv: 2307.09288 [cs.CL].

[16] J. Wei *et al.*, *Emergent abilities of large language models*, 2022. arXiv: 2206.07682 [cs.CL].

[17] O. Li, M. Subramanian, A. Saakyan, S. CH-Wang, and S. Muresan, *Normdial: A comparable bilingual synthetic dialog dataset for modeling social norm adherence and violation*, 2023. arXiv: 2310.14563 [cs.CL].

[18] Y. Fung, T. Chakrabarty, H. Guo, O. Rambow, S. Muresan, and H. Ji, "NORMSAGE: Multi-lingual multi-cultural norm discovery from conversations on-the-fly," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, H. Bouamor, J. Pino, and K. Bali, Eds., Singapore: Association for Computational Linguistics, Dec. 2023, pp. 15 217–15 230. DOI: 10.18653/v1/2023.emnlp-main.941. [Online]. Available: https://aclanthology.org/2023.emnlp-main.941.

[19] C. Ziems, J. Dwivedi-Yu, Y.-C. Wang, A. Halevy, and D. Yang, "NormBank: A knowledge bank of situational social norms," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds., Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 7756–7776. DOI: 10.18653/v1/2023.acl-long.429. [Online]. Available: https://aclanthology.org/2023.acl-long.429.

[20] V. Rawte, A. Sheth, and A. Das, *A survey of hallucination in large foundation models*, 2023. arXiv: 2309.05922 [cs.AI].

[21] E. Sheng, K.-W. Chang, P. Natarajan, and N. Peng, "The woman worked as a babysitter: On biases in language generation," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, K. Inui, J. Jiang, V. Ng, and X. Wan, Eds., Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 3407–3412. DOI: 10.18653/v1/D19-1339. [Online]. Available: https://aclanthology.org/D19-1339.

[22] L. Fan *et al.*, "In plain sight: Media bias through the lens of factual reporting," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 6343–6349. DOI: 10.18653/v1/D19-1664. [Online]. Available: https://www.aclweb.org/anthology/D19-1664.

[23] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423. [Online]. Available: https://www.aclweb.org/anthology/N19-1423.

[24] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," in *Advances in neural information processing systems*, 2019, pp. 5754–5764.

[25] Y. Liu *et al.*, "Roberta: A robustly optimized bert pretraining approach," *ArXiv*, vol. abs/1907.11692, 2019.

[26] M. E. Peters *et al.*, "Deep contextualized word representations," *arXiv preprint arXiv:1802.05365*, 2018.

[27] A. Vaswani *et al.*, "Attention is all you need," *CoRR*, vol. abs/1706.03762, 2017. arXiv: 1706.03762. [Online]. Available: http://arxiv.org/abs/1706.03762.

[28] F. Biessmann, "Automating political bias prediction," *arXiv preprint arXiv:1608.02195*, 2016.

[29] K. Johnson and D. Goldwasser, "Classification of moral foundations in microblog political discourse," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 720–730. DOI: 10.18653/v1/P18-1067. [Online]. Available: https://www.aclweb.org/anthology/P18-1067.

[30] K. Johnson and D. Goldwasser, "Identifying stance by analyzing political discourse on Twitter," in *Proceedings of the First Workshop on NLP and Computational Social Science*, D. Bamman *et al.*, Eds., Austin, Texas: Association for Computational Linguistics, Nov. 2016, pp. 66–75. DOI: 10.18653/v1/W16-5609. [Online]. Available: https://aclanthology.org/W16-5609.

[31] A. Kornilova, D. Argyle, and V. Eidelman, "Party matters: Enhancing legislative embeddings with author attributes for vote prediction," *CoRR*, vol. abs/1805.08182, 2018. arXiv: 1805.08182. [Online]. Available: http://arxiv.org/abs/1805.08182.

[32] S. Chen, D. Khashabi, W. Yin, C. Callison-Burch, and D. Roth, "Seeing Things from a Different Angle: Discovering Diverse Perspectives about Claims," in *Proc. of the Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2019. [Online]. Available: http://cogcomp.org/papers/CKYCR19.pdf.

[33] M. Iyyer, A. Guha, S. Chaturvedi, J. Boyd-Graber, and H. Daumé III, "Feuding families and former Friends: Unsupervised learning for dynamic fictional relationships," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, San Diego, California: Association for Computational Linguistics, Jun. 2016, pp. 1534–1544. DOI: 10.18653/v1/N16-1180. [Online]. Available: https://www.aclweb.org/anthology/N16-1180.

[34] X. Han, E. Choi, and C. Tan, "No permanent Friends or enemies: Tracking relationships between nations from news," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 1660–1676. DOI: 10.18653/v1/N19-1167. [Online]. Available: https://www.aclweb.org/anthology/N19-1167.

[35] D. Demszky *et al.*, "Analyzing polarization in social media: Method and application to tweets on 21 mass shootings," *arXiv preprint arXiv:1904.01596*, 2019.

[36] S. Roy and D. Goldwasser, "Weakly supervised learning of nuanced frames for analyzing polarization in news media," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, B. Webber, T. Cohn, Y. He, and Y. Liu, Eds., Online: Association for Computational Linguistics, Nov. 2020, pp. 7698–7716. DOI: 10.18653/v1/2020.emnlp-main.620. [Online]. Available: https://aclanthology.org/2020.emnlp-main.620.

[37] D. Diermeier, J.-F. Godbout, B. Yu, and S. Kaufmann, "Language and ideology in congress," *British Journal of Political Science*, vol. 42, no. 1, pp. 31–55, 2012. DOI: 10.1017/S0007123411000160.

[38] D. Preoiuc-Pietro, Y. Liu, D. Hopkins, and L. Ungar, "Beyond binary labels: Political ideology prediction of twitter users," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vancouver, Canada: Association for Computational Linguistics, Jul. 2017, pp. 729–740. DOI: 10.18653/v1/P17-1068. [Online]. Available: https://www.aclweb.org/anthology/P17-1068.

[39] V. Kulkarni, J. Ye, S. Skiena, and W. Y. Wang, "Multi-view models for political ideology detection of news articles," *CoRR*, vol. abs/1809.03485, 2018. arXiv: 1809.03485. [Online]. Available: http://arxiv.org/abs/1809.03485.

[40] J. Clinton, S. Jackman, and D. Rivers, "The statistical analysis of roll call data," *American Political Science Review - AMER POLIT SCI REV*, vol. 98, Apr. 2003. DOI: 10.1017/S0003055404001194.

[41] A. Kornilova, D. Argyle, and V. Eidelman, "Party matters: Enhancing legislative embeddings with author attributes for vote prediction," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, I. Gurevych and Y. Miyao, Eds., Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 510–515. DOI: 10.18653/v1/P18-2081. [Online]. Available: https://aclanthology.org/P18-2081.

[42] P. Patil *et al.*, "Roll call vote prediction with knowledge augmented models," in *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 574–581. DOI: 10.18653/v1/K19-1053. [Online]. Available: https://www.aclweb.org/anthology/K19-1053.

[43] G. Spell, B. Guay, S. Hillygus, and L. Carin, "An Embedding Model for Estimating Legislative Preferences from the Frequency and Sentiment of Tweets," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Online: Association for Computational Linguistics, Nov. 2020, pp. 627–641. DOI: 10.18653/v1/2020.emnlp-main.46. [Online]. Available: https://www.aclweb.org/anthology/2020.emnlp-main.46.

[44] M. Davoodi, E. Waltenburg, and D. Goldwasser, "Understanding the language of political agreement and disagreement in legislative texts," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds., Online: Association for Computational Linguistics, Jul. 2020, pp. 5358–5368. DOI: 10.18653/v1/2020.acl-main.476. [Online]. Available: https://aclanthology.org/2020.acl-main.476.

[45] S. Mohammad, S. Kiritchenko, P. Sobhani, X. Zhu, and C. Cherry, "A dataset for detecting stance in tweets," in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, Portoro, Slovenia: European Language Resources Association (ELRA), May 2016, pp. 3945–3952. [Online]. Available: https://www.aclweb.org/anthology/L16-1623.

[46] W. Fang, M. Nadeem, M. Mohtarami, and J. Glass, "Neural multi-task learning for stance prediction," in *Proceedings of the Second Workshop on Fact Extraction and VERification (FEVER)*, Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 13–19. DOI: 10.18653/v1/D19-6603. [Online]. Available: https://www.aclweb.org/anthology/D19-6603.

[47] C. Li and D. Goldwasser, "Encoding social information with graph convolutional networks forPolitical perspective detection in news media," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, A. Korhonen, D. Traum, and L. Màrquez, Eds., Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 2594–2604. DOI: 10.18653/v1/P19-1247. [Online]. Available: https://aclanthology.org/P19-1247.

[48] N. Pali, J. Vladika, D. ubeli, I. Lovreni, M. Buljan, and J. najder, "TakeLab at SemEval-2019 task 4: Hyperpartisan news detection," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA: Association for Computational Linguistics, Jun. 2019, pp. 995–998. DOI: 10.18653/v1/S19-2172. [Online]. Available: https://www.aclweb.org/anthology/S19-2172.

[49] R. Baly *et al.*, "What was written vs. who read it: News media profiling using text analysis and social media context," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds., Online: Association for Computational Linguistics, Jul. 2020, pp. 3364–3374. DOI: 10.18653/v1/2020.acl-main.308. [Online]. Available: https://aclanthology.org/2020.acl-main.308.

[50] P. Velikovi, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.

[51] T. Müller, F. Piccinno, M. Nicosia, P. Shaw, and Y. Altun, "Answering conversational questions on structured data without logical forms," *arXiv preprint arXiv:1908.11787*, 2019.

[52] K. T. Poole, H. Rosenthal, *et al.*, *Congress: A Political-economic History of Roll Call Voting*. Oxford University Press on Demand, 1997.

[53] S. M. Gerrish and D. M. Blei, "Predicting legislative roll calls from text," in *Proceedings of the 28th International Conference on Machine Learning, ICML 2011*, 2011.

[54] V.-A. Nguyen, J. Boyd-Graber, P. Resnik, and K. Miler, "Tea party in the house: A hierarchical ideal point topic model and its application to republican legislators in the 112th congress," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2015, pp. 1438–1448.

[55] P. Kraft, H. Jain, and A. M. Rush, "An embedding model for predicting roll-call votes," in *Proceedings of the 2016 conference on empirical methods in natural language processing*, 2016, pp. 2066–2070.

[56] A. Kornilova, D. Argyle, and V. Eidelman, "Party matters: Enhancing legislative embeddings with author attributes for vote prediction," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2018, pp. 510–515.

[57] G. Spell, B. Guay, S. Hillygus, and L. Carin, "An embedding model for estimating legislative preferences from the frequency and sentiment of tweets," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 627–641.

[58] J. Baumgartner, "Twitter Tweets for Donald J. Trump (@realdonaldtrump)," version V1, *Harvard Dataverse*, 2019. DOI: 10.7910/DVN/KJEBIL. [Online]. Available: https://doi.org/10.7910/DVN/KJEBIL.

[59] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky, "The stanford corenlp natural language processing toolkit," in *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, 2014, pp. 55–60.

[60] J. Brank, G. Leban, and M. Grobelnik, "Annotating documents with relevant wikipedia concepts," in *Proceedings of Slovenian KDD Conference on Data Mining and Data Warehouses (SiKDD)*, 2017.

[61] T. Wolf *et al.*, "Huggingface's transformers: State-of-the-art natural language processing," *ArXiv*, vol. abs/1910.03771, 2019.

[62] J. Vaes, M. P. Paladino, and C. Magagnotti, "The human message in politics: The impact of emotional slogans on subtle conformity," *The Journal of Social Psychology*, vol. 151, no. 2, pp. 162–179, 2011, PMID: 21476460. DOI: 10.1080/00224540903510829. eprint: https://doi.org/10.1080/00224540903510829. [Online]. Available: https://doi.org/10.1080/00224540903510829.

[63] D. Weber and F. Neumann, "Amplifying influence through coordinated behaviour in social networks," *Social Network Analysis and Mining*, vol. 11, 2021.

[64] K. Bach, "Pragmatics and the philosophy of language," in Wiley Online Library, Jan. 2008, pp. 463–487, ISBN: 9780470756959. DOI: 10.1002/9780470756959.ch21.

[65] E. M. Bender and A. Koller, "Climbing towards NLU: On meaning, form, and understanding in the age of data," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds., Online: Association for Computational Linguistics, Jul. 2020, pp. 5185–5198. DOI: 10.18653/v1/2020.acl-main.463. [Online]. Available: https://aclanthology.org/2020.acl-main.463.

[66] Y. Bisk *et al.*, "Experience grounds language," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, B. Webber, T. Cohn, Y. He, and Y. Liu, Eds., Online: Association for Computational Linguistics, Nov. 2020, pp. 8718–8735. DOI: 10.18653/v1/2020.emnlp-main.703. [Online]. Available: https://aclanthology.org/2020.emnlp-main.703.

[67] D. Fried, N. Tomlin, J. Hu, R. Patel, and A. Nematzadeh, "Pragmatics in language grounding: Phenomena, tasks, and modeling approaches," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds., Singapore: Association for Computational Linguistics, Dec. 2023, pp. 12 619–12 640. DOI: 10.18653/v1/2023.findings-emnlp.840. [Online]. Available: https://aclanthology.org/2023.findings-emnlp.840.

[68] J. Andreas and D. Klein, "Reasoning about pragmatics with neural listeners and speakers," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, J. Su, K. Duh, and X. Carreras, Eds., Austin, Texas: Association for Computational Linguistics, Nov. 2016, pp. 1173–1182. DOI: 10.18653/v1/D16-1125. [Online]. Available: https://aclanthology.org/D16-1125.

[69] P. Sharma, N. Ding, S. Goodman, and R. Soricut, "Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, I. Gurevych and Y. Miyao, Eds., Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 2556–2565. DOI: 10.18653/v1/P18-1238. [Online]. Available: https://aclanthology.org/P18-1238.

[70] M. Alikhani, P. Sharma, S. Li, R. Soricut, and M. Stone, "Cross-modal coherence modeling for caption generation," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds., Online: Association for Computational Linguistics, Jul. 2020, pp. 6525–6535. DOI: 10.18653/v1/2020.acl-main.583. [Online]. Available: https://aclanthology.org/2020.acl-main.583.

[71] S. I. Wang, P. Liang, and C. D. Manning, "Learning language games through interaction," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, K. Erk and N. A. Smith, Eds., Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 2368–2378. DOI: 10.18653/v1/P16-1224. [Online]. Available: https://aclanthology.org/P16-1224.

[72] A. Suhr *et al.*, "Executing instructions in situated collaborative interactions," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, K. Inui, J. Jiang, V. Ng, and X. Wan, Eds., Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 2119–2130. DOI: 10.18653/v1/D19-1218. [Online]. Available: https://aclanthology.org/D19-1218.

[73] R. Lachmy, V. Pyatkin, A. Manevich, and R. Tsarfaty, "Draw me a flower: Processing and grounding abstraction in natural language," *Transactions of the Association for Computational Linguistics*, vol. 10, B. Roark and A. Nenkova, Eds., pp. 1341–1356, 2022. DOI: 10.1162/tacl_a_00522. [Online]. Available: https://aclanthology.org/2022.tacl-1.77.

[74] C. Potts, "Goal-driven answers in the cards dialogue corpus," in *Proceedings of the 30th west coast conference on formal linguistics*, Cascadilla Proceedings Project Somerville, MA, 2012, pp. 1–20.

[75] T. Udagawa and A. Aizawa, "A natural language corpus of common grounding under continuous and partially-observable context," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 7120–7127.

[76] H. H. Clark and S. E. Brennan, "Grounding in communication.," *Perspectives on Socially Shared Cognition*, pp. 127–149, 1991.

[77] D. Traum, "A computational theory of grounding in natural language conversation," 1994.

[78] R. Stalnaker, "Common ground," *Linguistics and philosophy*, vol. 25, no. 5/6, pp. 701–721, 2002.

[79] S. Black *et al.*, "GPT-NeoX-20B: An open-source autoregressive language model," in *Proceedings of the ACL Workshop on Challenges & Perspectives in Creating Large Language Models*, 2022. [Online]. Available: https://arxiv.org/abs/2204.06745.

[80] R. Pujari and D. Goldwasser, "Understanding politics via contextualized discourse processing," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, M.-F. Moens, X. Huang, L. Specia, and S. W.-t. Yih, Eds., Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 1353–1367. DOI: 10.18653/v1/2021.emnlp-main.102. [Online]. Available: https://aclanthology.org/2021.emnlp-main.102.

[81] S. Feng *et al.*, "Par: Political actor representation learning with social context and expert knowledge," *arXiv preprint arXiv:2210.08362*, 2022.

[82] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, J. Burstein, C. Doran, and T. Solorio, Eds., Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423. [Online]. Available: https://aclanthology.org/N19-1423.

[83] Y. Liu *et al.*, "Roberta: A robustly optimized bert pretraining approach," *ArXiv*, vol. abs/1907.11692, 2019.

[84] T. Wolf *et al.*, "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Q. Liu and D. Schlangen, Eds., Online: Association for Computational Linguistics, Oct. 2020, pp. 38–45. DOI: 10.18653/v1/2020.emnlp-demos.6. [Online]. Available: https://aclanthology.org/2020.emnlp-demos.6.

[85] T. B. Brown *et al.*, "Language models are few-shot learners," *CoRR*, vol. abs/2005.14165, 2020. arXiv: 2005.14165. [Online]. Available: https://arxiv.org/abs/2005.14165.

[86] M. Sherif, *The psychology of social norms*, 1936.

[87] H. C. Triandis *et al.*, *Culture and social behavior*. McGraw-Hill New York, 1994.

[88] M. Finnemore, "Norms, culture, and world politics: Insights from sociology's institutionalism," *International organization*, vol. 50, no. 2, pp. 325–347, 1996.

[89] D. Hymes *et al.*, "Models of the interaction of language and social life," 1972.

[90] A. Srivastava, A. Rastogi, and A. R. et. al., *Beyond the imitation game: Quantifying and extrapolating the capabilities of language models*, 2023. arXiv: 2206.04615 [cs.CL].

[91] O.-M. Camburu, T. Rocktäschel, T. Lukasiewicz, and P. Blunsom, *E-snli: Natural language inference with natural language explanations*, 2018. arXiv: 1812.01193 [cs.CL].

[92] M. Suzgun *et al.*, *Challenging big-bench tasks and whether chain-of-thought can solve them*, 2022. arXiv: 2210.09261 [cs.CL].

[93] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa, *Large language models are zero-shot reasoners*, 2023. arXiv: 2205.11916 [cs.CL].

[94] B. Prystawski, M. Y. Li, and N. D. Goodman, *Why think step by step? reasoning emerges from the locality of experience*, 2023. arXiv: 2304.03843 [cs.AI].

[95] J. Wei *et al.*, "Chain of thought prompting elicits reasoning in large language models," *CoRR*, vol. abs/2201.11903, 2022. arXiv: 2201.11903. [Online]. Available: https://arxiv.org/abs/2201.11903.

[96] A. P. Hare, "Roles, relationships, and groups in organizations: Some conclusions and recommendations," *Small group research*, vol. 34, no. 2, pp. 123–154, 2003.

[97] R. C. Schank and R. P. Abelson, *Scripts, plans, goals, and understanding: An inquiry into human knowledge structures*. Psychology press, 1977.

[98] M. L. Pacheco, T. Islam, L. Ungar, M. Yin, and D. Goldwasser, "Interactive concept learning for uncovering latent themes in large text collections," in *Findings of the Association for Computational Linguistics: ACL 2023*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds., Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 5059–5080. DOI: 10.18653/v1/2023.findings-acl.313. [Online]. Available: https://aclanthology.org/2023.findings-acl.313.

[99] A. Smith, V. Kumar, J. Boyd-Graber, K. Seppi, and L. Findlater, "Closing the loop: User-centered design and evaluation of a human-in-the-loop topic modeling system," in *23rd International Conference on Intelligent User Interfaces*, 2018, pp. 293–304.

[100] K. Roberts, A. Dowell, and J.-B. Nie, "Attempting rigour and replicability in thematic analysis of qualitative research data; a case study of codebook development," *BMC medical research methodology*, vol. 19, pp. 1–8, 2019.

[101] S. Roy and D. Goldwasser, "Analysis of nuanced stances and sentiment towards entities of us politicians through the lens of moral foundation theory," in *Proceedings of the ninth international workshop on natural language processing for social media*, 2021, pp. 1–13.

[102] D. Demszky *et al.*, "Analyzing polarization in social media: Method and application to tweets on 21 mass shootings," *arXiv preprint arXiv:1904.01596*, 2019.

[103] R. Pujari, C. Wu, and D. Goldwasser, "We demand justice!: Towards social context grounding of political texts," in *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, Eds., Miami, Florida, USA: Association for Computational Linguistics, Nov. 2024, pp. 362–372. DOI: 10.18653/v1/2024.emnlp-main.22. [Online]. Available: %5Curl%7Bhttps://aclanthology.org/2024.emnlp-main.22/%7D.

[104] C.-H. Chiang and H.-y. Lee, "Can large language models be an alternative to human evaluations?" *arXiv preprint arXiv:2305.01937*, 2023.

[105] Z. Li, D. Park, J. Kiseleva, Y.-B. Kim, and S. Lee, "Deus: A data-driven approach to estimate user satisfaction in multi-turn dialogues," *arXiv preprint arXiv:2103.01287*, 2021.

[106] M. Bano, D. Zowghi, and J. Whittle, "Exploring qualitative research using llms," *arXiv preprint arXiv:2306.13298*, 2023.

[107] Y. Bang *et al.*, "A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity," *arXiv preprint arXiv:2302.04023*, 2023.

[108] S. Yao *et al.*, "React: Synergizing reasoning and acting in language models," *arXiv preprint arXiv:2210.03629*, 2022.

[109] D. Hendrycks *et al.*, "Aligning ai with shared human values," *arXiv preprint arXiv:2008.02275*, 2020.

[110] Q. Wu *et al.*, "Autogen: Enabling next-gen llm applications via multi-agent conversation framework," *arXiv preprint arXiv:2308.08155*, 2023.

[111] N. Arabzadeh *et al.*, "Assessing and verifying task utility in llm-powered applications," *arXiv preprint arXiv:2405.02178*, 2024.

[112] Y.-T. Chen, H.-H. Huang, and H.-H. Chen, "MPDD: A multi-party dialogue dataset for analysis of emotions and interpersonal relationships," English, in *Proceedings of the Twelfth Language Resources and Evaluation Conference*, N. Calzolari *et al.*, Eds., Marseille, France: European Language Resources Association, May 2020, pp. 610–614, ISBN: 979-10-95546-34-4. [Online]. Available: https://aclanthology.org/2020.lrec-1.76.

[113] Y. Chen *et al.*, *Cped: A large-scale chinese personalized and emotional dialogue dataset for conversational ai*, 2022. arXiv: 2205.14727 [cs.CL].

[114] A. Dubey *et al.*, "The llama 3 herd of models," *arXiv preprint arXiv:2407.21783*, 2024.

[115] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "Qlora: Efficient finetuning of quantized llms," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[116] K. R. Chandu, Y. Bisk, and A. W. Black, "Grounding 'grounding' in NLP," in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, C. Zong, F. Xia, W. Li, and R. Navigli, Eds., Online: Association for Computational Linguistics, Aug. 2021, pp. 4283–4305. DOI: 10.18653/v1/2021.findings-acl.375. [Online]. Available: https://aclanthology.org/2021.findings-acl.375.

[117] H. H. Clark, *Using language*. Cambridge university press, 1996.

[118] N. Arabzadeh and C. Clarke, "Fréchet distance for offline evaluation of information retrieval systems with sparse labels," in *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, Y. Graham and M. Purver, Eds., St. Julian's, Malta: Association for Computational Linguistics, Mar. 2024, pp. 420–431. [Online]. Available: https://aclanthology.org/2024.eacl-long.26.

[119] Y. Weng *et al.*, "Large language models are better reasoners with self-verification," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds., Singapore: Association for Computational Linguistics, Dec. 2023, pp. 2550–2575. DOI: 10.18653/v1/2023.findings-emnlp.167. [Online]. Available: https://aclanthology.org/2023.findings-emnlp.167.

[120] Y. Cui, W. Che, T. Liu, B. Qin, and Z. Yang, "Pre-training with whole word masking for chinese bert," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3504–3514, 2021.

[121] M. Wang *et al.*, "Deep graph library: A graph-centric, highly-performant package for graph neural networks," *arXiv preprint arXiv:1909.01315*, 2019.

[122] W. L. Hamilton, R. Ying, and J. Leskovec, *Inductive representation learning on large graphs*, 2018. arXiv: 1706.02216.

[123] Y. Liu, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[124] D. Hovy and S. Prabhumoye, "Five sources of bias in natural language processing," *Language and linguistics compass*, vol. 15, no. 8, e12432, 2021.

[125] M. Geva, Y. Goldberg, and J. Berant, "Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets," *arXiv preprint arXiv:1908.07898*, 2019.

[126] S. Gautam and M. Srinath, "Blind spots and biases: Exploring the role of annotator cognitive biases in nlp," *arXiv preprint arXiv:2404.19071*, 2024.

[127] T. Davidson, D. Bhattacharya, and I. Weber, "Racial bias in hate speech and abusive language detection datasets," *arXiv preprint arXiv:1905.12516*, 2019.

[128] A. Das *et al.*, "Investigating annotator bias in large language models for hate speech detection," *arXiv preprint arXiv:2406.11109*, 2024.

[129] J. Doughman and W. Khreich, "Gender bias in text: Labeled datasets and lexicons," *arXiv preprint arXiv:2201.08675*, 2022.

[130] Y. Chang *et al.*, "A survey on evaluation of large language models," *ACM transactions on intelligent systems and technology*, vol. 15, no. 3, pp. 1–45, 2024.

[131] OpenAI, *Gpt-4o technical report*, Accessed: 2025-05-27, 2024. [Online]. Available: https://openai.com/index/gpt-4o-system-card/.

[132] G. DeepMind, *Gemini 2.5*, Accessed: 2025-05-27, 2025. [Online]. Available: https://deepmind.google/discover/blog/gemini-25-our-world-leading-model-is-getting-even-better/.

[133] Anthropic, *Claude 4 models*, Accessed: 2025-05-27, 2025. [Online]. Available: https://www.anthropic.com/news/claude-4.

[134] MetaAI, *Llama 4 herd of models*, Accessed: 2025-05-27, 2025. [Online]. Available: https://ai.meta.com/blog/llama-4-multimodal-intelligence/.

[135] A. Liu *et al.*, "Deepseek-v3 technical report," *arXiv preprint arXiv:2412.19437*, 2024.

[136] R. Pujari, C. Wu, and D. Goldwasser, "we demand justice!: Towards social context grounding of political texts," in *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, Eds., Miami, Florida, USA: Association for Computational Linguistics, Nov. 2024, pp. 362–372. DOI: 10.18653/v1/2024.emnlp-main.22. [Online]. Available: https://aclanthology.org/2024.emnlp-main.22/.

[137] R. Pujari and D. Goldwasser, "LLM-human pipeline for cultural grounding of conversations," in *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, L. Chiruzzo, A. Ritter, and L. Wang, Eds., Albuquerque, New Mexico: Association for Computational Linguistics, Apr. 2025, pp. 1029–1048, ISBN: 979-8-89176-189-6. [Online]. Available: https://aclanthology.org/2025.naacl-long.48/.

[138] N. Mehta and D. Goldwasser, "An interactive framework for profiling news media sources," in *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, K. Duh, H. Gomez, and S. Bethard, Eds., Mexico City, Mexico: Association for Computational Linguistics, Jun. 2024, pp. 40–58. DOI: 10.18653/v1/2024.naacl-long.3. [Online]. Available: https://aclanthology.org/2024.naacl-long.3/.

[139] S. Roy, "Weakly supervised characterization of discourses on social and political movements on online media," Ph.D. dissertation, Purdue University Graduate School, 2023.

[140] N. Mehta, "A framework to identify online communities for social media analysis," Ph.D. dissertation, Purdue University Graduate School, 2024.

[141] N. R. Crick and K. A. Dodge, "A review and reformulation of social information-processing mechanisms in children's social adjustment.," *Psychological bulletin*, vol. 115, no. 1, p. 74, 1994.

[142] Z. Ke *et al.*, *A survey of frontiers in llm reasoning: Inference scaling, learning to reason, and agentic systems*, 2025. arXiv: 2504.09037 [cs.AI]. [Online]. Available: https://arxiv.org/abs/2504.09037.

[143] C. Ziems, W. Held, O. Shaikh, J. Chen, Z. Zhang, and D. Yang, *Can large language models transform computational social science?* 2024. arXiv: 2305.03514 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2305.03514.

[144] H. Peters and S. Matz, *Large language models can infer psychological dispositions of social media users*, 2024. arXiv: 2309.08631 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2309.08631.

[145] A. Vaswani *et al.*, *Attention Is All You Need*, arXiv:1706.03762 [cs], Aug. 2023. [Online]. Available: http://arxiv.org/abs/1706.03762.

[146] Y. Gao *et al.*, *Retrieval-augmented generation for large language models: A survey*, 2024. arXiv: 2312.10997 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2312.10997.

VITA

[Put a brief autobiographical sketch here.]

INDEX